

Tomi Isomäki

Festo Tripod EXPT -rinnakkaisrakenteisen robotin ohjelmointi, suunnittelu ja käyttöönotto

Metropolia Ammattikorkeakoulu
Insinööri (AMK)
Automaatiotekniikka
Insinöörityö
4.6.2012

Tekijä Otsikko Sivumäärä Aika	Tomi Isomäki Festo Tripod EXPT -rinnakkaisrakenteisen robotin suunnittelu, ohjelmointi ja käyttöönotto 45 sivua + 4 liitettä 4.6.2012
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Automaatiotekniikka
Suuntautumisvaihtoehto	Kappaletavaratekniikka
Ohjaajat	Yliopettaja Jouni Jokelainen Tuotemarkkinointijohtaja Jukka Kenttämies
<p>Tämä insinöörityö käsittelee Feston rinnakkaisrakenteisen teollisuusrobottisoluun tehtävää esittelysovellusta ja sen käyttöönottoa. Työ tehtiin osittain valmiina olevaan robottisoluun. Ensimmäiseksi työssä tutustutaan Feston rinnakkaisrakenteisen robotin ominaisuuksiin ja rakenteisiin. Projektiin liittyen työssä käydään läpi robotin asennuksen, käyttöönoton, suunnittelun ja ohjelmoinnin kannalta projektin keskeisimpiä asioita.</p> <p>Työssä suunniteltiin robotille työalustaa, jolle tämä rakentaa puisen radan. Robotti ajaa puista autoa rataa pitkin. Työalustan tarkoitus oli pitää puiset raiteet oikeassa paikassa niin, että robotti pystyisi tarttumaan niihin ja viemään ne takaisin omalle paikalleen.</p> <p>Työllä saavutettiin toimiva puinen junaratasovellus. Työalustalla saatiin ratkaistua keskeinen ongelma, miten raiteet saadaan pysymään paikoillaan robotin ajaessa autoa radalla. Tämän avulla myös robotin kolaritilanteet saatiin minimoitua. Työ käyttöönotettiin Tallinnan teknillisessä korkeakoulussa, jossa robotin käyttäjä opastettiin robotin uudelleen käyttöön ottamiseen sähköjen häviämisen tai virhetilanteen jälkeen.</p>	
Avainsanat	rinnakkaisrakenteinen, robotti, tripod, ohjelmointi

Author Title Number of Pages Date	Tomi Isomäki Designing, programming and commissioning Festo Tripod EXPT -parallel structured robot 45 pages + 4 appendices 4 June 2012
Degree	Automation Technology
Degree Programme	Bachelor of Engineering
Specialisation option	Manufacturing Automation
Instructors	Jouni Jokelainen, Principal Lecturer Jukka Kenttämies, Product Marketing Manager
<p>This thesis presents the making and commissioning of a demonstration application for Festo's parallel robot industrial robotic cell. The work was partially ready for the robot cell. First, this study looks at Festo parallel robot's features and structures. Relating to the project to thesis, the robot installation, commissioning, planning and programming of the project's key issues are gone through.</p> <p>During the project a work surface was planned for the robot and a wooden track was built by the robot. The robot drives a wooden car along the track. The purpose of project surface was to keep the wooden rails at the right place, so that the robot would be able to grasp them and bring them back in their place.</p> <p>A wooden railway line application was achieved in the project. The work surface solved a key problem of how to rails will remain in place while the robot is driving a car on the track. This also helped to minimize any robot crash situations. The project was commissioned in Tallinn University of Technology, where the robot user was guided to re-commissioning the robot in case of a power cut or an error in function.</p>	
Keywords	parallel, robot, tripod, programming

Sisällys

1	Johdanto	1
1.1	Työn tarkoitus ja tavoitteet	1
1.2	Yritys	1
2	Festo Tripod EXPT -robotti	2
2.1	Robotin yleiskuvaus	2
2.2	Ohjausjärjestelmä	2
2.2.1	Keskusyksikkö	3
2.2.2	Operointipaneeli	6
2.2.3	Käsiohjain	7
2.3	Robotti	9
2.4	Suorituskyky	10
2.5	Mekaaninen rakenne	13
2.6	Koordinaatisto	14
3	Järjestelmän suunnittelu	15
3.1	Lähtötilanne	15
3.2	Tartunnan suunnittelu	15
3.2.1	Ongelman kuvaus	17
3.2.2	Työalustan suunnittelu	17
3.2.3	Työalustan toteutus	20
3.3	Pneumaattinen suunnittelu	22
3.3.1	Lähtötilanne	22
3.3.2	Paineensäädin	23
3.3.3	Pneumaattisesti toimiva mekaaninen tarttuja	23
3.3.4	Alipainekomponentit	24
3.4	Sähkösuunnittelu	26
3.4.1	Ohjainnapit	26
3.4.2	Hätäpysäytyslaitteet	26

3.4.3	I/O:n kortit ja riviliittimet	28
3.4.4	Anturit	29
4	Ohjelmointi	30
4.1	Ohjelman rakenne	30
4.2	Ohjelmointi	32
4.2.1	Muuttujat	32
4.2.2	Perusohjelmointi	33
4.2.3	Lasketut pisteet	34
4.2.4	Opettamalla ohjelmointi	34
5	Käyttöönotto	35
5.1	Toiminnan kuvaus	35
5.2	Ohjeiden laatiminen	41
5.3	Testaus	41
5.4	Käyttöönotto asiakkaidentiloissa	41
6	Yhteenveto	42
	Lähteet	44
	Liitteet	
	Liite 1. Työalustan layout	
	Liite 2. Pneumatiikkakuva	
	Liite 3. Sähkökuvat	
	Liite 4. Käyttöohjeet	

1 Johdanto

1.1 Työn tarkoitus ja tavoitteet

Tämän insinööri työn aiheena on Festo Tripod EXPT -rinnakkaisrakenteisen robotin käyttöönotto, suunnittelu ja ohjelmointi.

Opinnäytetyö on tarkoitus toteuttaa valmiina olevaan demo robottiprojektiin, josta puuttuu toimiva demosovellus. Projektissa asennetaan robotille sähköisiä ja pneumaattisia instrumentteja sekä suunnitellaan tähän työalusta.

Tämä insinöörityö käsittelee yleisesti robotiikkaa. Projektin osalta opinnäytetyössä käydään läpi Festo Tripod EXPT teknisiä tietoja, robotin ohjelmointia, 3D-mallinnusta, demosovelluksen laatimista ja käyttöohjeiden tekoa.

Työn tarkoituksena on saada asennettua ja ohjelmoitua robottisoluun näyttävä ja luotettava junaratasovellus. Robotti rakentaa ympyränmuotoisen radan puisista kiskoista ja ajaa tässä puista autoa. Työn valmistuttua pystytään kyseisellä robottisolulla näyttämään asiakkaille robotin toimintaa erilaisissa messu- ja esittelytilaisuuksissa.

1.2 Yritys

Työ tehdään Festo Oy:ssä Vantaan Koivuhaassa. Festo on yksi maailman johtavista automaatioteknologian toimittajista ja kouluttajista, joka pyrkii tuottamaan erilaisia ratkaisuja tehdas- ja prosessiautomaatioon ympäri maailman. [1.]

2 Festo Tripod EXPT -robotti

2.1 Robotin yleiskuvaus

Robotti on monipuolinen käsittelylaite, joka voi tehdä täysin tai osittain tietokoneen ohjaamana mekaanisia operaatioita. Robotilla on oltava vähintään kolme uudelleen ohjelmoitavissa olevaa akselia. Uudelleen ohjelmointi on yksi tärkeimpiä piirteitä, mutta nykyaikaisissa robottisovelluksissa on tärkeää saada myös tietoa erilaisten antureiden ja toimintaympäristön tapahtumien perusteella ja muuttaa toimintaa näiden mukaan [2, s.13; 4, s.16; 5 s.2-3].

Teollisuusrobottimalleja on maailmassa erilaisia, ja niitä suunnitellaan jatkuvasti lisää erilaisten patenttien ja sovellusten takia [2, s.12]. Tässä työssä keskitytään pelkästään Festo:n rinnakkaisrakenteisen Tripod EXPT -teollisuusrobotin rakenteeseen ja ominaisuuksiin, koska se on oleellinen osa tätä projektia.

Robotti koostuu kahdesta tyypillisestä teollisuusrobotin pääkomponentista, ohjausjärjestelmästä, joka ohjaa robotin mekaanisia liikkeitä, ja manipulaattorista, joka toteuttaa ohjausjärjestelmältä tulevat käskyt mekaanisiksi liikkeiksi. [2, s.13; 3; 4 s.16.]

2.2 Ohjausjärjestelmä

Robotin ohjausjärjestelmä koostuu yleensä keskusyksiköstä, massamuistista, missä ohjelmat pystytään säilyttämään, käyttöpaneelistä, käsiohjaimesta ja ulkoisista liitännöistä, joilla robottia pystytään operoimaan ja ohjelmoimaan, nivelkohtaisista servolaitteista, joilla servomoottoreita ohjataan, ja teholähteistä, jotka muuttavat sähköä laitteille sopivaksi. [2, s.34.]

Tässä projektissa käytettävä ohjausjärjestelmä on esitetty kuvassa 1. Se on rakennettu kompaktiin kokoon, jotta se mahtuisi demolaitteen sisään.



Kuva 1. Robotin ohjausjärjestelmä edestä

Lisäksi ohjausjärjestelmä sisältää servo-ohjaimet, jotka näkyvät kuvassa 2. Servo-ohjaimet on sijoitettu asennuslevyn taakse tilanpuutteen vuoksi.



Kuva 2. Servo-ohjaimet

2.2.1 Keskusyksikkö

Tämän robotin ohjausjärjestelmän ns. aivot ovat Feston CMRX-C1-robottiohjain (kuva 3). Robottiohjain suorittaa siihen ohjelmoitua ohjelmaa, jonka perusteella se ohjaa liikeohjaimia ja muita ulkopuolisia laitteita, esim. magneettiventtiileitä ja kontaktoreita, sekä käsittelee erilaisilta antureilta saatuja tietoja.

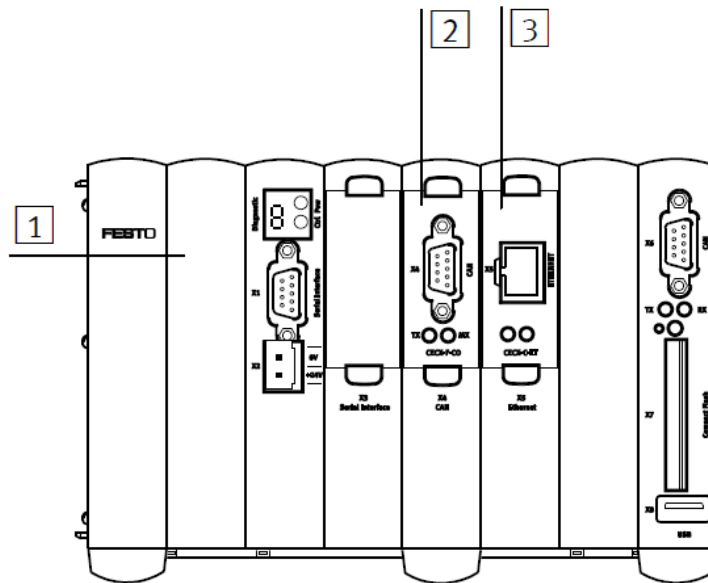
Kyseinen robottiohjain on rakennettu modulaariseksi, joten se voidaan rakentaa halutunlaiseksi [10, s.18].



Kuva 3. CMRX-C1 -robottiohjain [7.]

Ohjain koostuu kolmesta moduulista: keskusohjausyksiköistä, valinnaisesta moduulista ja oheislaitemoduulista.

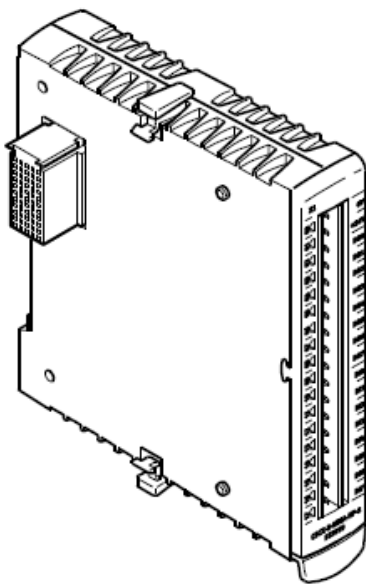
Keskusohjausyksikkö (kuva 4) sisältää 400 Mhz kellotaajuudella toimivan prosessorin, korttipaikan optiomoduuleille, compact flash -muistikortille, oheislaitemoduuleille, CAN-liitännän servoakseleille ja USB-liitännän. [10, s.18]



Kuva 4. Keskusohjausyksikkö (1), joka sisältää kaksi valinnaismoduulia (2) ja (3) [10, s.26]

Vallinnaismoduuleille on kolme valmista paikkaa keskusohjausyksikössä, mihin ne voidaan haluttaessa lisätä. Yllä olevassa kuvassa 4 on lisätty kaksi valinnaista moduulia: Ethernet-liitäntä Ethernet-ohjainta varten ja CAN-liitäntä CAN-ohjainta varten. [10, s26.]

Oheislaitemuoduleita voidaan lisätä keskusohjausmoduulin oikealle puolelle, niin kuin kuvassa 3 on tehty. Kuvassa 5 on yksittäinen digital I/O kortti. [10, s.18]



Kuva 5. Digital input/output-oheislaitemuuli CECX-D-8E8A-NP-2

2.2.2 Operointipaneeli

Robotinohjaimessa olevan ns. operointipaneelin (kuva 6) avulla voidaan valita robotin haluttu toimintatila HAND tai AUTO. Lisäksi tämä sisältää suojapiirin ja hätä-seis-kytkimien kuittauksen ja akselikohtaiset vapautuskytkimet.



Kuva 6. Robotin operointipaneeli

AUTO (Automaattinen toiminta)

Tätä toimintoa käytetään, jos halutaan robotin suorittavan tälle laadittua työohjelmaa. Tällöin robotin AUTO-merkkivalo palaa, käsiohjaus ei ole mahdollista ja turvatoimenpiteet ovat aktiivisina. [4, s.30]

HAND (Käsiohjaustoiminta)

Tätä toimintoa käytetään, jos halutaan liikuttaa robottia manuaalisesti haluttuun paikkaan, testata robotin ohjelman toimintaa tai opettaa siihen uusia paikkatietoja. Tällöin robotin HAND-merkkivalo palaa, liikenopeusnopeus on maksimissaan 250 mm/s ja turvatoimenpiteet ovat osittain ohitettut. [4, s.30]

BREMSE LÖSEN (Jarrun vapautus)

Näillä kytkimillä voidaan vapauttaa akselikohtaisesti niiden jarrut, jolloin robottia pystytään liikuttamaan käsin ilman moottoreita haluttuun paikkaan. Tätä toimintoa käytetään yleensä robottiin kohdistuvissa huoltotoimenpiteissä. [5, s.35]

NOT – AUS QUITTIEREN (Hätä-seis-kuittaus)

Tästä napista voidaan kuitata hätä-seis-piiri, mikäli hätä-seis-kytkimet ovat vapautettut.

SHUTZTÜREN QUITTIEREN (Suojapiirin kuittaus)

Tästä napista voidaan kuitata suojapiiri, mikäli suojapiirin kytkimet ovat kiinni.

2.2.3 Käsiohjain

Tämä ohjauskeskus sisältää Feston CDSA-käsiohjaimen (kuva 7), jolla voidaan kommunikoida robotin kanssa. Se sisältää kaiken oleellisen, mitä robotin liikuttamiseen tai säätämiseen tulee.

Käsiohjaimella pystytään mm. ohjaamaan robottia akselikohtaisesti tai ohjelmallisesti, ohjelmoimaan sille pisteitä koordinaatistosta opettamalla, lukemaan diagnostiikkaa käytön aikana ja säätämään nopeuksia.



Kuva 7. Robotin käsiohjain Festo CDSA

Kuvassa 8 on esitetty käsiohjaimen komponenttien merkitykset.



Kuva 8. Festo CDSA-käsiohjaimen painikkeet ja komponentit [8]

1. Start- ja Stop-painikkeet
2. Häätä-seis-painike
3. Run-, Error-, Motion- ja Process-ledit
4. Funktioiden valintanapit
5. 6,5" TFT-kosketusnäyttö
6. Liiketoimintojen liikutusnapit
7. Kosketuskynä
8. Sallintakytkin eli ns. kuolleenmiehenkytkin
9. Kaapelin ulostulot

2.3 Robotti

Festo Tripod EXPT-robotti kuuluu standardin ISO 8373 määritelmän mukaan rinnakkaisrakenteisiin robotteihin [2, s.12]. Se koostuu kolmesta Feston DGE25-lineaariohjaimesta ja EMMS-servomoottorista, jotka on kiinnitetty toisiinsa rinnan ja täten muodostavat robottiin pyramidi muotoisen rakenteen. Tällaista rakennetta kutsutaan suljetuksi kinematiikaksi, joka mahdollista kevyemmän rakenteen sekä kolminkertaisen nopeuden ja jäykemmän rakenteen karteesisiiin robotteihin verrattuna [2, s.16; 9].

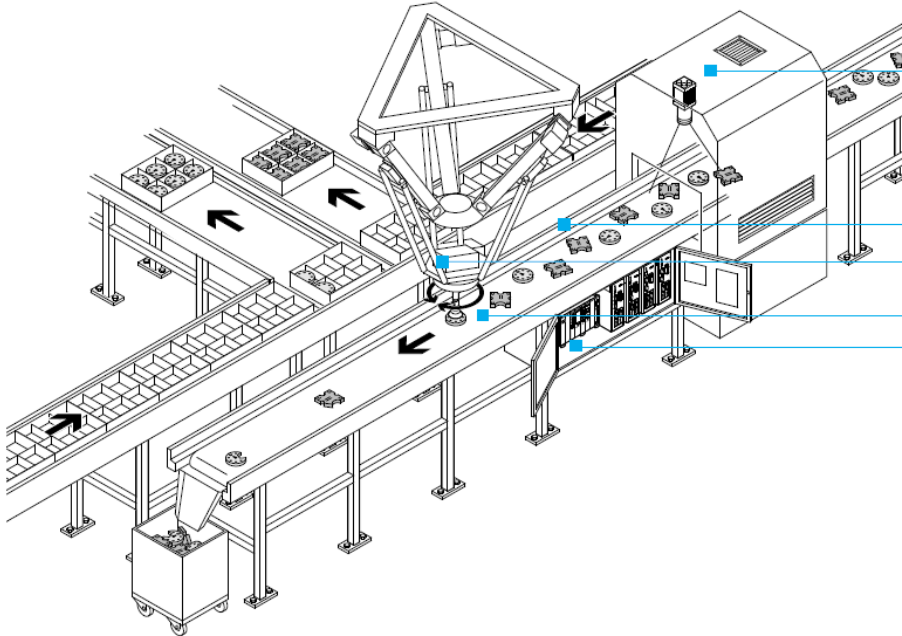


Kuva 9. Festo Tripod EXPT -robotti [11]

Tripod EXPT-robottia käytetään teollisuudessa seuraaviin sovelluksiin:

- Poimi ja sijoita (Pick and Place)
- Liimaus (Bonding)
- Merkintä (Labelling)
- Paletointi (Palletizing)
- Lajittelu (Sorting)
- Ryhmittely (Grouping)
- Erottaminen (Separaiting)
- Kiinnittäminen (Mounting)

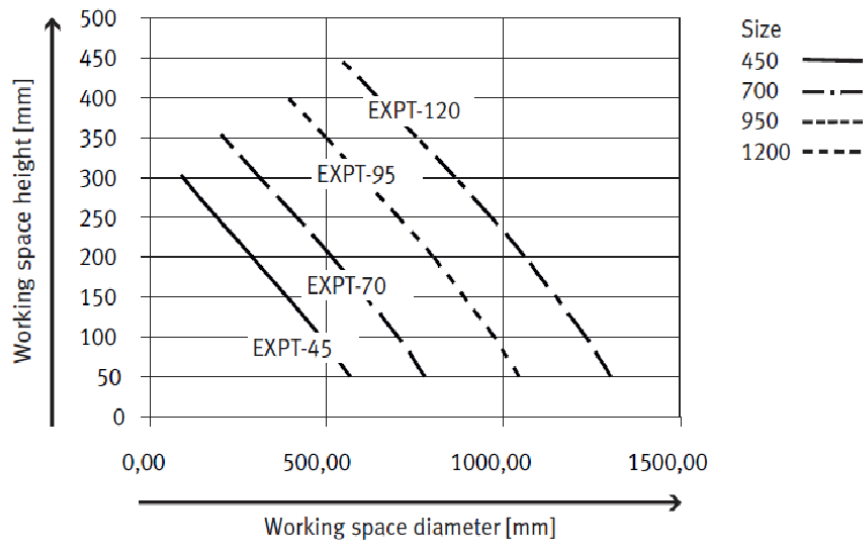
Kuvassa 10 on tyypillinen esimerkki pick and place -sovelluksesta, jossa kuljettimelta poimitaan kevyt komponentti ja sijoitetaan se tässä tapauksessa laatikkoon, jossa se lähtee eteenpäin.



Kuva 10. Pick and place -sovellusesimerkki [10]

2.4 Suorituskyky

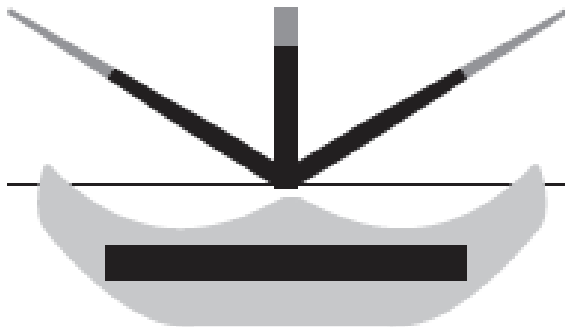
Tässä projektissa käytettävä robotti on prototyyppi eli se on ensimmäisiä malleja, joita myöhemmin on tullut lisää. Niissä ominaisuudet ovat hieman muuttuneet. Malliltaan ja ominaisuuksiltaan robotti vastaa EXPT-45:tä. Kuvasta 11 käy ilmi robotti -mallit sekä näiden työskentelyalueet.



Kuva 11. Tripod EXPT -robottien mallit sekä näiden työskentelyalueet. [14, s63]

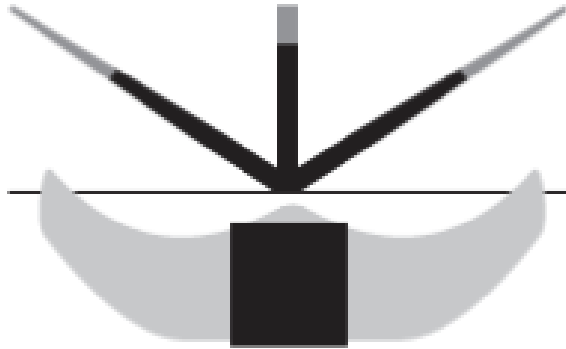
Robotin työskentelyalue on melko rajallinen. Sivusta katsottuna sen työskentelyalue näyttää muodoltaan rauskulta, alue kapenee sivuille mentäessä sekä kohoaa ylöspäin. Kuva 12 ja kuva 13 selventävät alueen kuvausta.

Kuvassa 12 käy ilmi, että mitä leveämpi alue, sitä pienempi on robotin työskentelyalue pystysuunnassa.



Kuva 12. Robotin työskentelyalue [14, s.63]

Kuvatsa 13 voidaan huomata, että mitä kapeampi työskentelyalue, sitä suurempi on työskentelyalue pystysuunnassa.



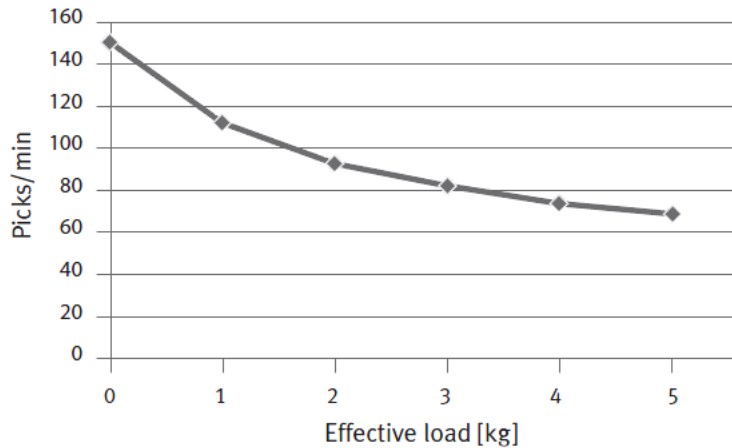
Kuva 13. Robotin työskentelyalue [14, s.63]

Taulukko 1:stä selviävät robotin tekniset tiedot. Arvot ovat samat robotin koosta ja mallista riippumatta.

Taulukko 1. Tripod EXPT-robotin tekniset tiedot

Kiihtyvyys	Nopeus	Kantavuus	Toistotarkkuus	Ratatarkkuus	Absoluuttinen tarkkuus
100 m/s ²	6 m/s	5 kg	±0,1 mm	±0,3 mm	±0,5 mm

Kantavuus tarkoittaa maksimipainoa, mitä robotti pystyy liikuttamaan. Kyseisessä robotissa liikutettava massa vaikuttaa syklin kierrosaikaan eli siihen, kuinka nopeasti robotti pystyy poimimaan. Kuvasta 14 nähdään eripainoisten massan vaikutus sykli aikaan.



Kuva 14. Painon vaikutus sykli aikaan [10, s.18]

Toistotarkkuus tarkoittaa tarkkuutta, millä robotin työkalupiste palaa takaisin aikaisemmin määrättyyn pisteeseen [3].

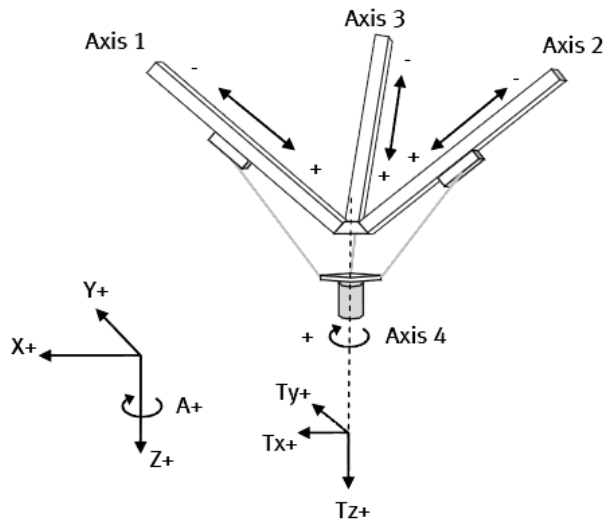
Ratatarkkuus tarkoittaa suurinta eroa liikkeelle kohtisuorissa suunnissa, nimellisen liikeradan ja saavutetun liikeratojen keskiarvojen välillä [2, s.182].

Absoluuttinen tarkkuus tarkoittaa tarkkuutta, millä robotti saadaan haluttuun pisteeseen robottiin sidotussa koordinaatistossa [3].

2.5 Mekaaninen rakenne

Robotti koostuu kolmesta lineaariohjaimesta, Axis 1, Axis 2 ja Axis 3, jotka on kytketty toisiinsa rinnan, niin kuin kuvasta 15 käy ilmi. Tällaista rakennetta kutsutaan suljetun kinematiikan järjestelmäksi [15, s.80].

Tässä kinematiikkajärjestelmässä on kolme vapausastetta sekä lisäksi työkalulaippaan lisätty kiertyvä työkalu Axis 4 [15, s.80].



Kuva 15. Robotin rakenne ja akselien liikesuunnat koordinaatistossa [15, s.80]

2.6 Koordinaatisto

Kuvassa 15 nähdään kolme koordinaatistoa, mitä robotti voi käyttää. Ne ovat Base-, World- ja Tool-koordinaatistot. Base-koordinaatisto eli peruskoordinaatisto on sidottu robotin akselien nollapisteeseen. Jos robotin liike halutaan määrittää peruskoordinaatistossa, on tälle myös määriteltävä tarttujan asento. Tool-koordinaatisto eli työkalukoordinaatisto on määritelty koordinaatistoon työkalupisteen avulla. Työkalupiste määritellään työkalun koon ja mallin mukaan. World-koordinaatisto eli maailmakoordinaatisto on sidottu robotin työskentely ympäristöön jonka avulla robotti voi mukautua liikkumisen aikana tehtävään työhön [5, s.16; 3; 15 s.58].

3 Järjestelmän suunnittelu

3.1 Lähtötilanne

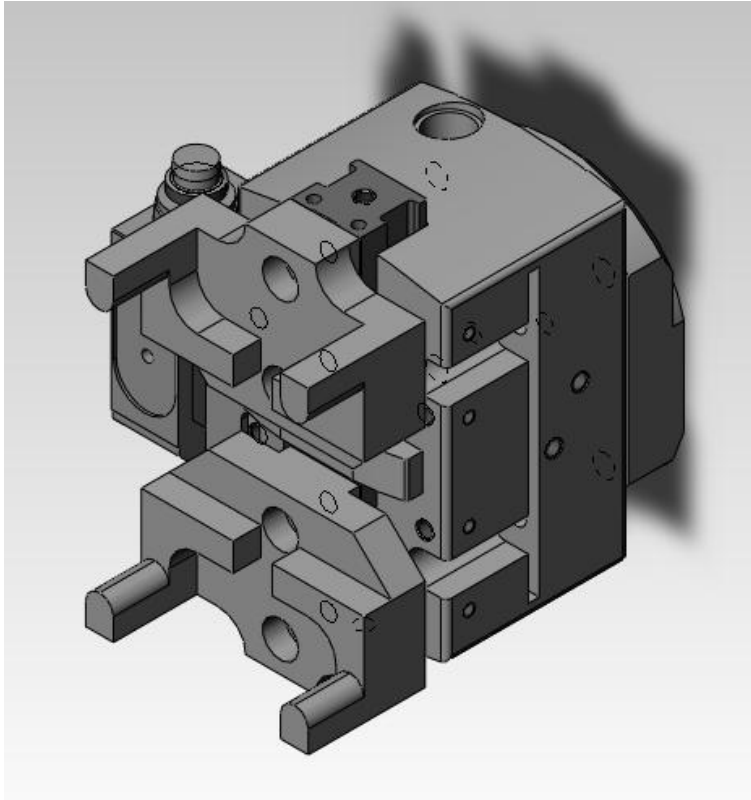
Työ alkoi itse robottiin toimintaan tutustumisella sekä testauksella. Lisäksi sain tietoa robottiin tehdyistä edellisistä sovelluksista ja niihin liittyvistä ongelmista. Tehtäväksi sain toteuttaa robotille sovelluksen, missä se rakentaa kahdeksasta puisesta kiskosta ympyrän muotoisen radan, jossa robotti ajaa autoa. Projektin sisältöön sovittiin myös seuraavia asioita:

- ohjelman helposti muunneltavuus
- tarttujan kiinnittyminen
- radan rakentamisessa syntyvien riskien kartoitus ja niiden poisto
- virhekoodien luonti sekä virhevalo
- start-nappäimen ja -valon lisäys käynnistämisen helpottamiseksi
- pikaohje robotin käyttöönottamiseen
- käyttöönottokoulutus.

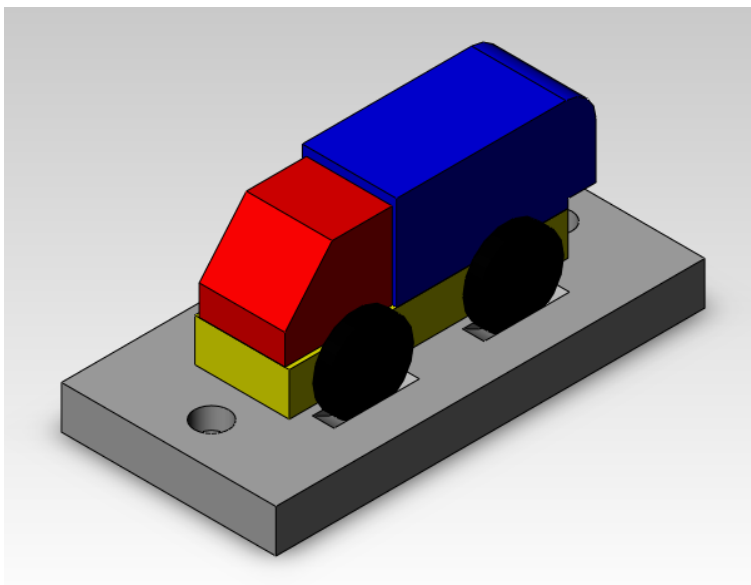
Robotin junaratasovelluksen luontia yrityksessä on kokeiltu aikaisemmin, mutta projektin paneutumiseen ei ole ollut riittävästi aikaa. Tämän projektin tarkoitus on tuoda yrityksen investoimalle robotille lisäarvoa sekä näyttävyyttä messu-, esittely- ja koulutustilaisuuksiin. Projektin pääpainona on sovelluksen varmatoimivuus sekä uudelleen käyttöönottaminen mahdollisimman helposti, koska näissä on ilmennyt ongelmia osien siirtymisen takia työalustalla.

3.2 Tartunnan suunnittelu

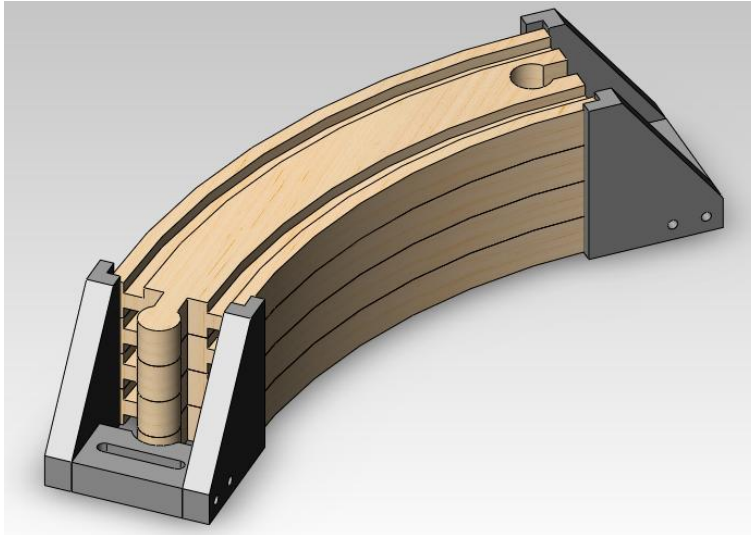
Robotin alkuperäisessä junaratasovelluksessa rata tehtiin robotin alkuperäiselle alustalle, jonka pintamateriaali oli lyhytkarvaista mattoa. Sovellukseen oli jo edellisellä kerralla suunniteltu robottiin tarttuja (kuva 16) sekä auton ja palikoiden makasiinit (kuvat 17 ja 18).



Kuva 16. Pneumaattinen tarttuja



Kuva 17. Auton makasiini



Kuva 18. Raiteiden makasiini

3.2.1 Ongelman kuvaus

Robotin solun liikuteltavuus sekä vähäinen anturointi tuo paljon haasteita teollisuuden robottisolun verrattuna, jossa robotin paikka ja työalusta eivät liiku, ja mikäli liikkuvat, on näissä riittävät anturoinnit, esim. konenäkö.

Solua liikuteltaessa junaraiteiden makasiinit tai robotin koordinaatisto voivat liikkua millimetrejä, mikä voi pahimmassa tapauksessa vaikuttaa sovellukseen junaraiteiden, junan tai robotin hajoamisen. Kyseisessä sovelluksessa käytettävät puiset junaraiteet eivät olleet tasalaatuisia, mikä junarataa kasatessa tuotti sovitusongelmia.

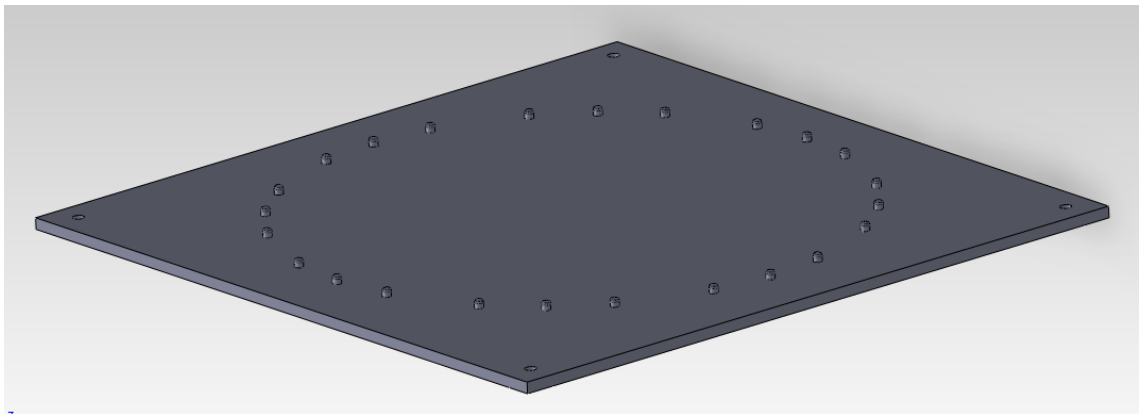
3.2.2 Työalustan suunnittelu

Työssä päätettiin käyttää jo valmiina olevaa tarttujaa kuva 16 ja makasiineja autolle ja raiteille. Robotille raiteiden tartunnan suunnittelua vaikeutti robotin työskentelyalue, joka oli robotin mallista johtuen pieni. Robotin liikealueista johtuen tartunnan korkeus sai olla korkeintaan 23 mm, joka todettiin useilla käytännön testeillä. Pääidea radan rakentamiselle oli toteuttaa rata alumiinilevylle, jossa tartunnat raiteisiin olisivat.

Tartunnan toteuttamiseksi tutkittiin seuraavia vaihtoehtoja:

- kohdistustappi
- magneettinen kiinnitys
- ura raiteille
- pneumaattinen kiinnitys imukupilla.

Ensimmäinen idea toteuttaa raiteiden kiinnitys oli tehdä näille levy, missä on kolme DIN 6325-lieriösokkaa yhtä raidetta kohti (kuva 19).



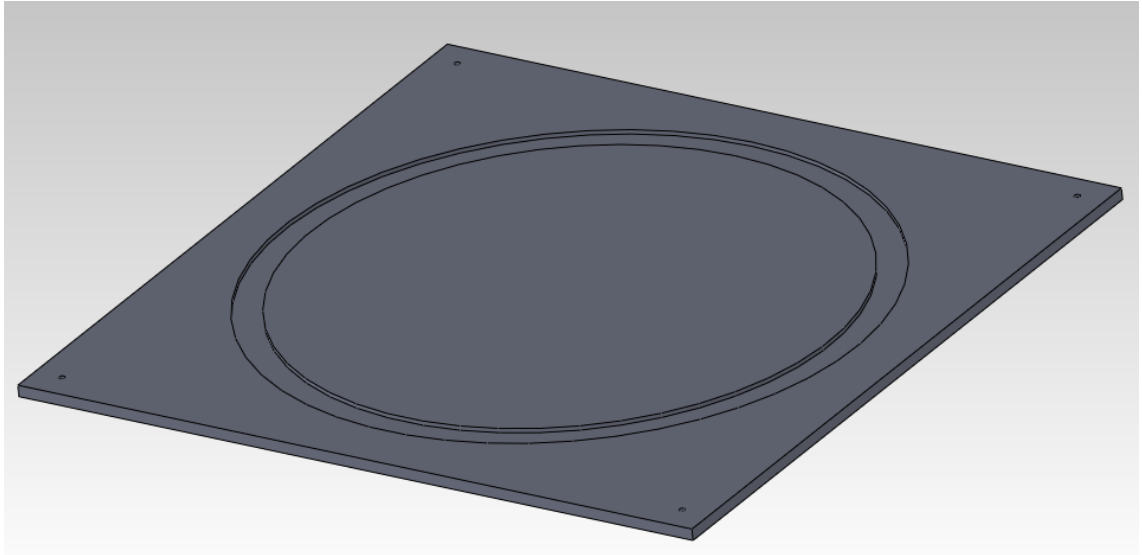
Kuva 19. Kohdistustapeilla varustettu työalusta

Ongelmana tässä ideassa olisi ollut raiteisiin porattavien reikien oikea kohta sekä reikien riittävä toleranssi. Raiteet on valmistettu puusta, mikä saattaa olla herkkä ilman kosteudelle sekä lämpötilalle, mistä syystä reikien koot olisivat muuttuneet.

Toinen idea oli käyttää sähkömagneettisia tarttujia, jotka olisi sijoitettu levyyn. Raiteisiin olisi laitettu pienet metallin palat, mihin magneetti olisi tarttunut.

Ongelma tässä ideassa oli sopivien magneettien löytäminen ja mahdolliset magneettiset häiriöt ympäristöön.

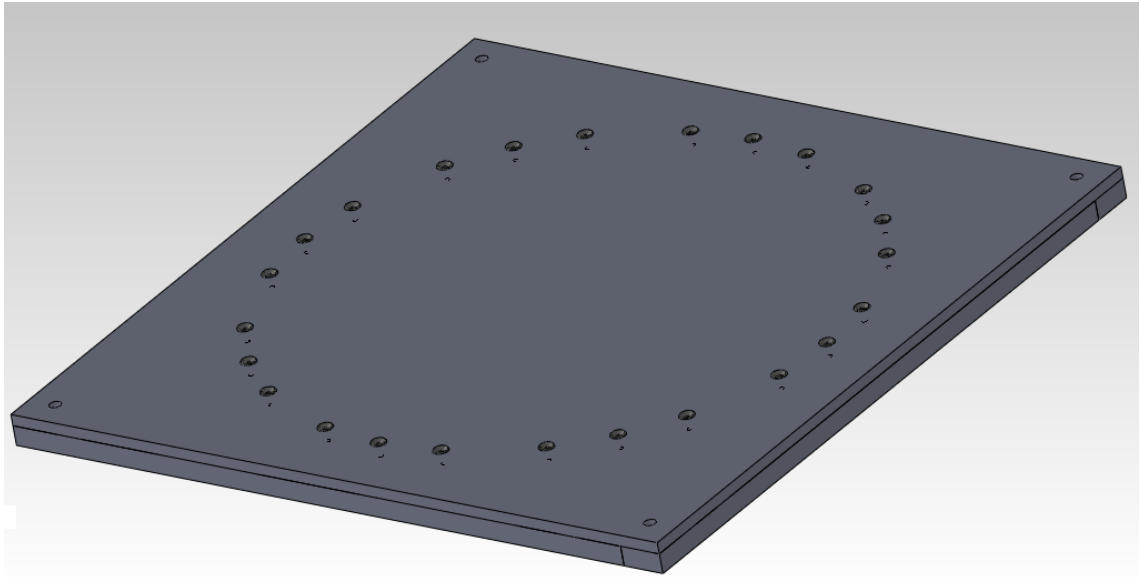
Kolmas idea oli tehdä raiteille oma ympyränmuotoinen ura levyyn, mihin rata rakennettaisiin (kuva 20).



Kuva 20. Uralla varustettu työalusta

Ongelma tässä ideassa oli näin suurelle alueelle tehtävän jyrsinnän tekevän yrityksen löytäminen ja levyn työstön hinta.

Neljäs idea oli käyttää Festo:n omia imukuppeja sekä ejektoreita. Koska tuotteet ovat Feston valmistamia, olisi näiden hankkiminen edullista. Työalusta koostuisi 1-3 imukuppitarttujasta jokaista raidetta kohti (kuva 21).



Kuva 21. Imukupeilla varustettu työalusta

3.2.3 Työalustan toteutus

Imukupeilla varustettu työalusta (kuva 21) valittiin toteutettavaksi, koska tämä oli edullisin ja esittelytilaisuuksissa voidaan esitellä myös muita Feston pneumatiikka-komponentteja. Työssä haasteellisin osuus oli löytää komponentit, jotka sopivat levyn alle niin, ettei levy noussut liian korkealle. Liitteessä 1 on alustan layout.

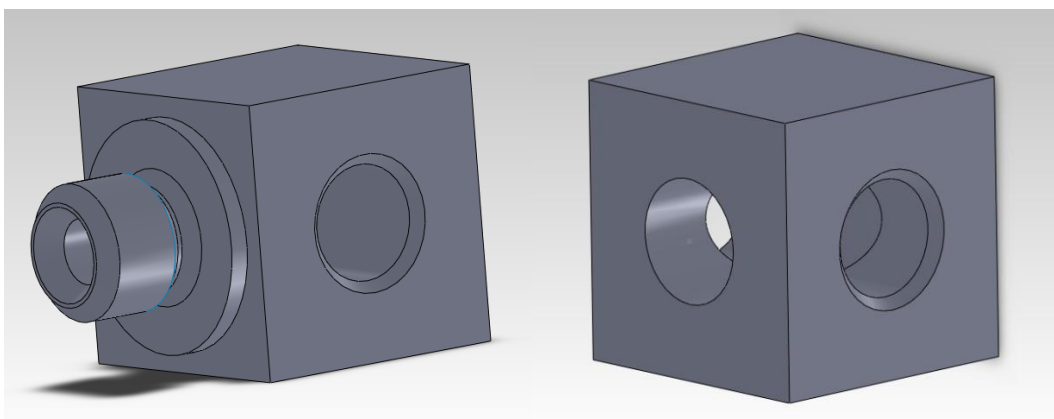
Työ alkoi sopivien imukuppien etsimisellä ja testaamisella. Testaukset aloitettiin testaamalla yhtiössä esittelylaatikossa olevilla imukupeilla. Testauksissa mitattiin alipainetta imukupin ollessa tarttunut junaraiteeseen sekä vertaamalla katalogissa oleviin pitovoimiin. Ihanteellinen alipaine olisi n. -0,95 bar. Testimittauksissa päästiin n. -0,85 bar tasoon, joka on kappaleisiin nähden hyvä, koska kappaleet olivat huokoista materiaalia. Taulukossa 2 on imukuppien koot, pitovoimat sekä omat kommentoinnit imukupin valintaa varten.

Taulukko 2. Imukuppien testituloksia

Imukupin koko (mm)	Pitovoima (N)	Kommentti
5,2	1,95	Vaatisi vähintään 8 kpl
4x10	2,6	Tuo korkeutta levyyn ja vaatisi vähintään kaksi kpl
10x30	19,6	Erinomainen pitävyys, mutta tuo korkeutta levyyn
10	5,5	Hyvä pitävyys vasta kolmella imukupilla

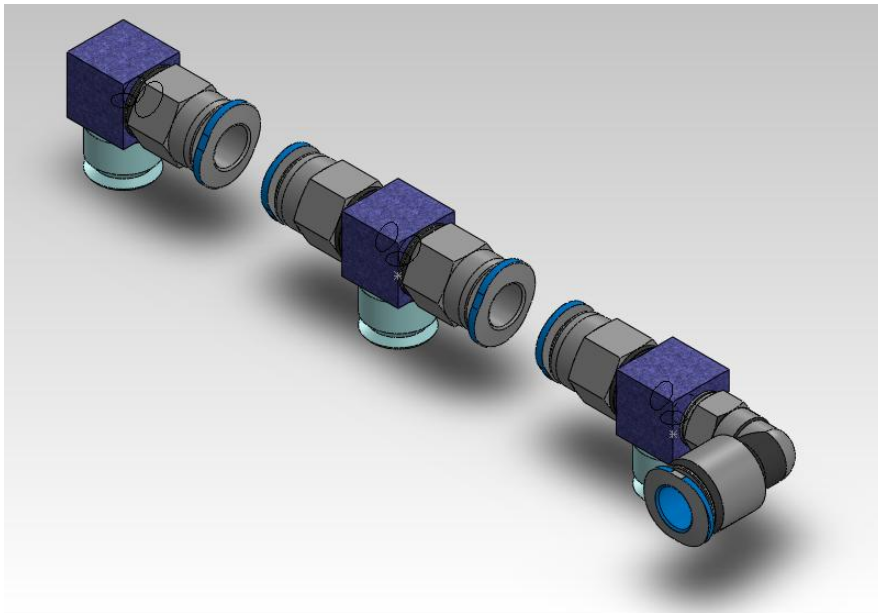
Imukupeista päädyttiin valitsemaan 10 mm halkaisijaltaan oleva imukuppi, joita sijoitettaisiin 3 kpl per raide. Ovaalin muotoinen imukuppi olisi ollut varteen otettava vaihtoehto, mutta se olisi tuottanut asennusvaikeuksia muotonsa ja korkeutensa vuoksi. Kolmella 10 mm halkaisijaltaan olevalta imukupilla päästään lähelle samaan pitovoimaan kuin 10x30 mm ovaalin muotoisella imukupilla.

Haastavin osuus työssä oli löytää imukupille sopiva kiinnitys sekä liitäntä. Feston tuotteista löytyi FR-M5-A/I-jatkoliitin (kuvassa 22 oikealla). Liittimestä piti poistaa M5 kierretappi ja tilalle piti tehdä reikä, jossa oli M4-kierre, johon imukuppi kierretään kiinni (kuvassa 22 vasemmalla). Liitin oli 10 mm leveä ja korkea, joten se mahtui hyvin levyn alle.



Kuva 22. Vasemmalla Feston FR-M5-A/I-kulmaliitin ja oikealla modifioitu FR-M5 A/I-kulmaliitin jossa M5-kierretappi on poistettu ja laitettu tilalle M4-reikä

Liittimiä tuli 3 kutakin raidetta kohti. Raiteen kiinnitys liittimet yhdistettiin pneumatiikkaan 6 mm:n PUN-letkulla ja QSM-6-M5-liittimillä (kuva 23).



Kuva 23. Imukupitarttujan kasaus

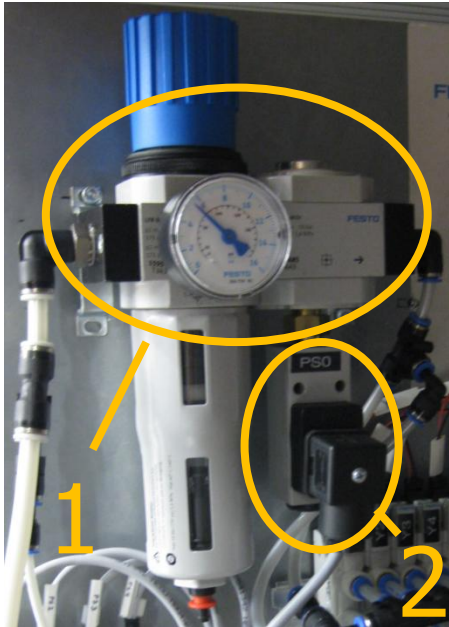
3.3 Pneumaattinen suunnittelu

3.3.1 Lähtötilanne

Robottijärjestelmä sisälsi ennestään pneumaattisen tarttujan HGPP-12, joka kytkettiin suoraan paineilmajärjestelmään 8 mm letkulla. Järjestelmään lisättiin pneumatiikkakomponentteja, mistä syystä oli hyvä piirtää näistä pneumatiikkakaavio (liite 2)

3.3.2 Paineensäädin

Järjestelmään lisättiin Feston huoltolaiteyksikkö, joka sisälsi säätöventtiilin ja painekytkimen (kuva 24).



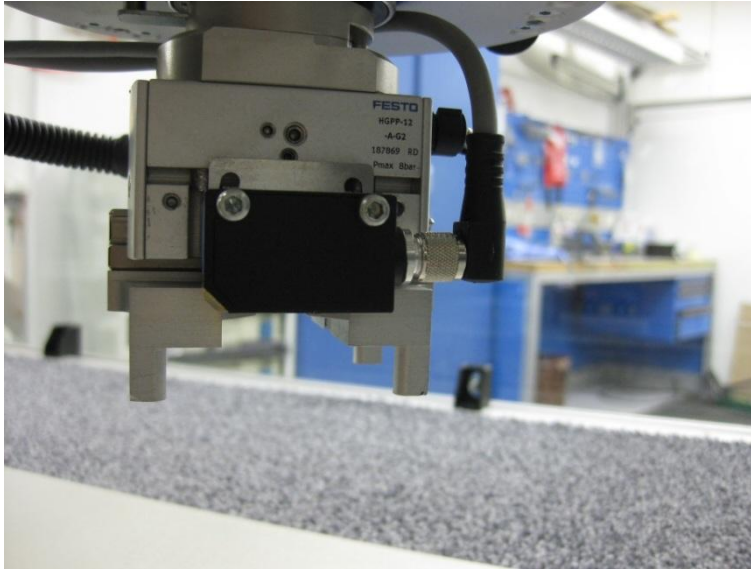
Kuva 24. Huoltolaite paineensäätimellä (1) ja -kytkimellä (2)

Paineensäätimellä säädettiin tulopaine imukuppien ejektoreille, jotka muodostavat alipaineen imukupeille. Ejektoreilla saadaan paras alipainetaso n. 4-5 bar paineella, jonka takia näiden syöttöpaine alennettiin paineensäätimellä.

Painekytkin on edellisessä kuvassa 24 kohde 2. Painekytkin on säädetty 4 bariin, joka on painetaso, jolla järjestelmä toimii. Paineen laskiessa alle 4 bar:n saa logiikka tästä tiedon digitaaliseen tuloon (DI), ja ohjelman suorittaminen loppuu.

3.3.3 Pneumaattisesti toimiva mekaaninen tarttuja

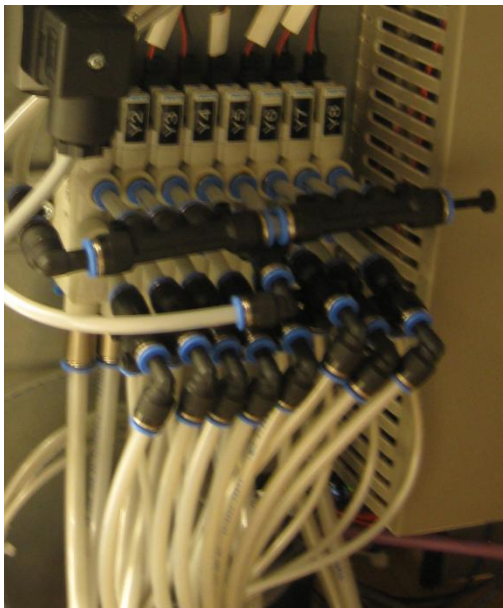
Järjestelmässä pidettiin vanha pneumaattinen tarttuja HGPP-12 (kuva 25), joka oli tähän suunniteltu ja joka toimi tässä järjestelmässä hyvin. Tarttuja toimisi pienemmälläkin paineella, mutta tässä järjestelmässä sille oli kytketty 6 bar käyttöpainetta. Tarttujaa ohjattiin pienellä magneettiventtiilillä, joka oli kytketty logiikan digitaaliseen lähtöön (DO).



Kuva 25. Festo HGPP-12 -tarttuja

3.3.4 Alipaineekomponentit

Junaradan tartunnassa käytetään Feston imukuppeja ESS-10 ja näihin alipainetta tuottamaan VN07-H-ejektoreita (kuva 26). Ejektorit olivat H-mallisia eli ne tuottavat korkean alipaineen, joka oli tässä projektissa tärkeä palikoiden kiinni pysymisen kannalta. Imukupit liitettiin 6 mm letkulla ejektoreihin, jotta päästäisiin vähimmillä painehäviöillä. Äänen vähentämiseksi ejektoreiden poistoliitännät yhdistettiin säiliöön, jossa oli päässä äänenvaimennin (kuva 27).



Kuva 26. VN07-H-ejektorit



Kuva 27. Ejektoreiden poisto säiliön kautta

Imukuppien rinnalle liitettiin Feston SDE3-alipaineanturit (kuva 28), joilla pystyttiin tarkkailemaan alipainetasoa. Antureissa oli kaksi kanavaa, joten yhteen anturiin pystyttiin liittämään kaksi tarkkailtavaa alipainetta. Antureiden asetuspaineeeksi asetettiin -0,65 bar, jonka alittaessa anturi antaa tiedon logiikan digitaaliseen tuloon (DI) ja pysäyttää ohjelman.



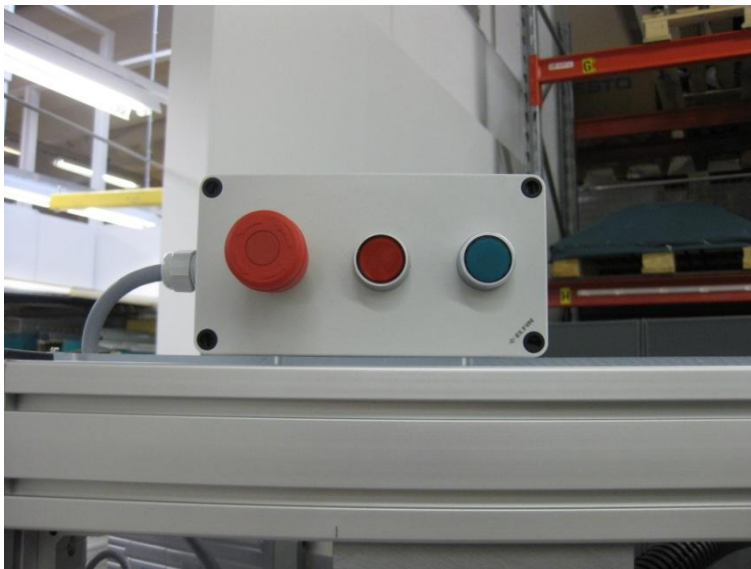
Kuva 28. SDE3-alipaineanturit

3.4 Sähkösuunnittelu

Robottisolu oli sähköistetty hyvin pitkälle, mutta johdot menivät hirveässä sotkussa, ja joitakin johtoja oli kytketty väärin. Tavoitteena oli saada johdot menemään siististi kourussa ja saada tarvittavat komponentit lisättyä ja kytkettyä sähköjärjestelmään.

3.4.1 Ohjainnapit

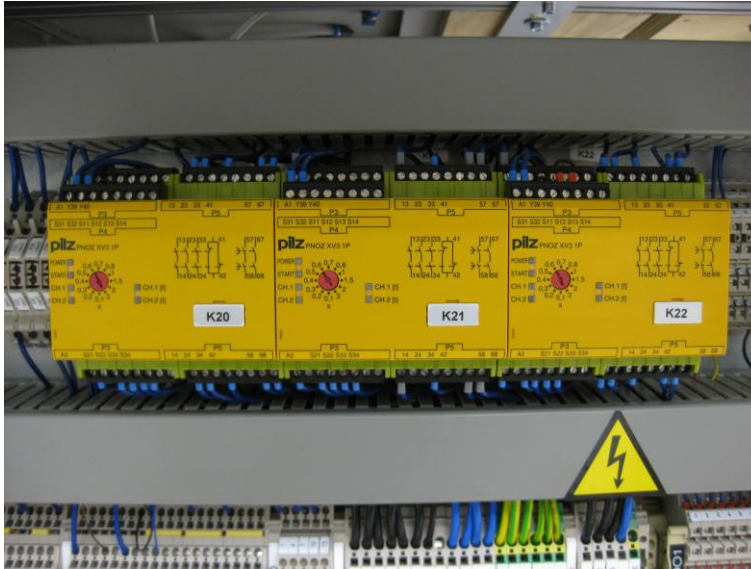
Robottia tullaan käyttämään erilaisissa esittelytilaisuuksissa, mistä syystä järjestelmään haluttiin lisätä Start- ja HS- (häätä-seis) -näppäimet sekä virhevalo käytön helpottamiseksi. Napit sijoitettiin pieneen koteloon robottisolun päälle (kuva 29).



Kuva 29. HS- ja Start-näppäimet sekä virhevalo.

3.4.2 Hätätyslaitteet

Järjestelmä sisälsi entuudestaan yhden HS-painikkeen sekä turvarajakytkimen. HS-painike sijaitsi käsiohjaimessa. Ongelma oli se, että jos käsiohjainta pidetään kaapin sisällä, menettää se silloin merkityksensä, mistä syystä painike päätettiin sijoittaa näkyvälle paikalle robottisolun päälle (kuva 29). HS-painike kytkettiin turvareleille (kuva 30) muiden hätätyslaitteiden kanssa sarjaan (liite 3).



Kuva 30. Turvareleet

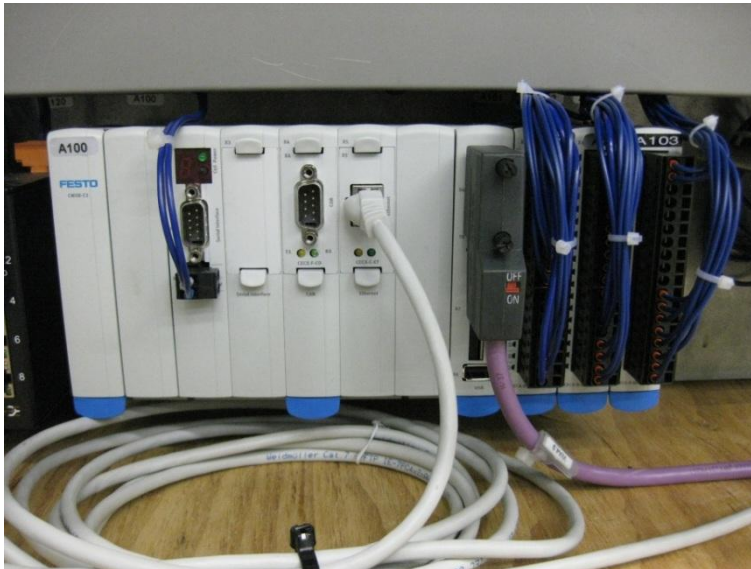
Järjestelmä sisälsi myös turvarajakytkimen (kuva 31), joka on laitettu estämään robotin työskentelytilaan pääseminen sen käytön aikana. Turvarajakytkin liitettiin turvareleisiin muiden turvakytkimien kanssa sarjaan (liite 3).



Kuva 31. Turvarajakytkin

3.4.3 I/O:n kortit ja riviliittimet

Järjestelmä sisälsi CMRX-C1 -robottiohjaimen (kuva 32), jossa oli kiinni 2 kpl I/O-kortteja eli yhteensä 16 kpl lähtö- ja tuloliitäntää. Järjestelmään piti kuitenkin lisätä yksi kortti lisää, jotta työalustan imukuppien alipainetieto ja ejektoreiden ohjaus pystyttäisiin lisäämään järjestelmään. Tarkemmat tiedot sähkökuissa (liite 3).



Kuva 32. CMRX-C1-robottiohjain

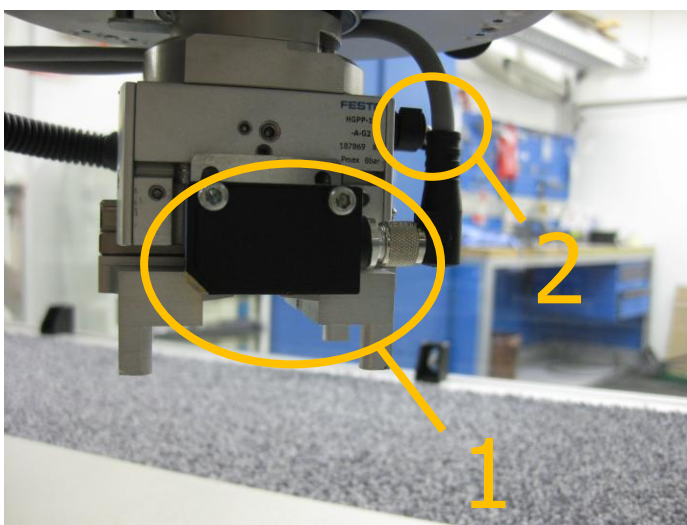
Järjestelmän päivittämisen helpottamiseksi järjestelmään lisättiin viisi eri ryhmää kolmi-kerrosriviliitimiä (kuva 33), joihin kytkettiin robottiohjaimen tulot ja lähdöt sekä 24 voltin plus- ja miinus-navat. Liitteessä 3 on kytkentäkuvat tästä.



Kuva 33. Lisätyt riviliittimet

3.4.4 Anturit

Tarttujaan oli asennettu jo valmiiksi pinnasta tunteva optinen anturi sekä asema-anturi (kuva 34). Optiselle anturille pystyttiin opettamaan pinta, mistä anturi tunnistaa kappaleen.



Kuva 34. Tarttujassa olevat optinen anturi (1) ja asema-anturi (2)

Asema-anturi (kuva 34 kohta 2) oli tarttujan sisällä, mistä se tunnisti tarttujan sormien asennon. Anturille pystytettiin opettamaan Feston SMH-AE1-arviointiyksikön (kuva 35) avulla kolmen erikappaleen koot, esim. auton, raiteiden ja täysin kiinni olevan tarttujan.



Kuva 35. Asema-anturin säätöyksikkö

4 Ohjelmointi

Robottia pystytään ohjelmoimaan erillisellä tietokoneella käyttäen siinä erillistä ohjelmointityökalua tai käsiohjaimella siinä olevia ohjelmointityökaluja. Käsiohjaimella ohjelmointi on paljon hitaampaa ja työläämpää kuin tietokoneella ohjelmointi, mistä syystä tämäkin projekti tehtiin tietokoneessa olevalla ohjelmointityökalulla. Järjestelmä sisälsi yksinkertaisia ohjelmia, joilla oli testattu robottia. Robotti ohjelmoitiin Feston FCT (Festo Configuration Tool) -ohjelmalla, jossa käytettiin Feston kehittämää FTL (Festo Teach Language) -ohjelmointikieltä. Ohjelmointikieli oli tekstipohjasta liikeohjelmointia, joka muistutti Pascal-ohjelmointikieltä.

4.1 Ohjelman rakenne

Ohjelmointi alkoi perusrakenteen suunnittelusta eli siitä, mitä muuttujia (variables), aliohjelmia (subprogram) itse ohjelma sisälsi ja missä järjestyksessä ja millä tyylillä

niitä pääohjelmassa (mainprogram) suoritettaisiin. Ohjelmassa oli yksi pääohjelma, jossa kutsuttiin aliohjelmia suorituksen mukaan. Tällaisessa rakenteessa etuna on, että koodin rakenteen ymmärtäminen ja seuraaminen on helpompaa kuin jos ohjelma olisi tehty yhteen ohjelmaan.

Kyseisen projektin ohjelma hierarkia oli seuraavanlainen:

- main
 - o calibration
 - o driving_circular
 - o driving_train
 - o gripping
 - o magazine
 - o rails_positions
 - o error

Aliohjelmat ovat kaikki edellä mainitun listan main-pääohjelman alla olevat ohjelmat, joita kutsutaan pääohjelmassa.

Aliohjelmien sisältö ja tehtävät:

Calibration eli kalibrointiohjelma. Sen päätarkoitus on saada robotin työalusta kohdalleen tai tarkistaa, jos se on kuljetuksessa tai jostain muusta syystä väärässä kohdassa. Pääohjelma kutsuu tätä aliohjelmaa alussa, minkä jälkeen se suorittaa kalibroinnin eli ajaa työlevyn reunaa pitkin, josta voidaan huomata, jos levy on liikkunut.

Driving_circular eli ympyrän ajo -ohjelma. Se ajaa ympyrän muotoista rataa maksimi-nopeudella. Tällä on tarkoitus näyttää, kuinka kovaa robotilla pystytään ajamaan.

Driving_train eli junan ajo -ohjelma. Tällä ohjelmalla ajetaan tarttujan avulla junaa sen radalla.

Gripping eli tartunta-ohjelma. Tämä ohjelma sisältää kaikki tartuntapisteet eli missä kohtaa tartutaan raiteisiin ja junaan.

Magazine eli makasiini-ohjelma. Tämä ohjelma sisältää kaikki makasiinissa olevat ns. varastopaikat palikoille sekä niiden lähestymispisteet.

Rails_positions eli raiteiden paikka -ohjelma. Ohjelmassa on laskettu raiteiden paikat ympyrässä sekä näiden lähestymispisteet, minkä avulla raiteet osataan viedä oikeaan paikkaan, kun ympyrän muotoista rataa rakennetaan.

Error eli vika-ohjelma. Tähän ohjelmaan on luotu vikaviestejä, mitä voi tulla itse pääohjelmasta ajettaessa. Tämä ohjelma pysäyttää robotin, jos pääohjelmasta suorittaessa sattuu jonkin virhe. Lisää tietoa vikaviesteistä saa liitteessä 4 olevasta käyttöohjeesta.

4.2 Ohjelmointi

4.2.1 Muuttujat

Muuttujia (variables) käytetään muuttujien, sijaintipisteiden, laskureiden, ajastimien määrittämiseen. Muuttujille voidaan antaa oletusarvot, joita ohjelma voi halutessaan muuttaa suorittaessaan tehtävää, esim. sijaintitietodon.

Esimerkki muuttujien määrittelykoodista:

```
calib_pos1:CARTPOS:=(-228.199,-215.014,158.0)
Speed0: REAL:= 80
Rail_pos: DINT:= 0
```

Muuttujille määritellään aina nimi ja muoto. Muoto voi olla esim. REAL eli reaaliluku, DINT eli kokonaisluku, BOOL eli bitti. Jos halutaan määrittää paikkatieto, pitää paikkatiedolle määrittää myös nimi, koordinaatisto AXISPOS eli akselikoordinaatisto, jossa määritellään akselien arvot tai CARTPOS eli karteellinen koordinaatisto, jossa määritellään X:n, Y:n ja Z:n arvot.

4.2.2 Perusohjelmointi

Feston FTL-ohjelmointikieli koostuu liikekäskyistä, joiden avulla robotti osaa mennä haluttuun paikkaan. Liikekäskyt sisältävät myös tiedon tavasta, miten robotti siirtyy pisteestä toiseen esim. ympyrämäisellä, lineaarisella, pisteestä pisteeseen-liikkeellä ja muilla näiden liikekäskyjen variaatiolla. Ennen liikekäskyn antamista robotille voidaan asettaa myös haluttu nopeus, kiihtyvyys tai suorittaa esim. IF- tai LOOP-funktio, jonka perusteella robotti tekee liikkeen.

Esimerkkikoodi perusliikekäskyistä:

```
IF pos.z > 104 THEN
pos.z := 104
Lin(pos)
END_IF

Vel(dynCart, Speed3)
Ptp(apupos)
Lin(abovepos)
CircIp(Train_ip_pos1,Train_end_pos1)
```

4.2.3 Lasketut pisteet

Jotta radan rakennuspaikkaa olisi helppo muuttaa, tehtiin ohjelma täysin lasketuilla pisteillä. Ainoastaan kotipiste oli määritelty, jonka mukaan muut pisteet laskettiin.

Esimerkkikoodi lasketuista pisteistä:

```
xc:= 201.40
yc:= 201.40
zc:= 40
abovepos:=startpos
abovepos.x:=abovepos.x+SIN(45)*xc
abovepos.y:=abovepos.y-SIN(45)*yc
abovepos.a:=abovepos.a-99
abovepos.z:=abovepos.z+zc
Lin(abovepos)
```

Edellä olevasta esimerkkikoodissa lasketaan junanraiteen 2. lähestymispisteen paikka kotipisteestä, minkä jälkeen siirytään lähestymispisteeseen.

4.2.4 Opettamalla ohjelmointi

Opettamalla ohjelmointi on tavalliseen ohjelmointiin verrattuna helppo tapa ohjelmoida robotti suorittamaan haluttu liike. Robotin työkalu viedään haluttuun paikkaan ja tallennetaan paikkatieto käsiohjaimella robottiohjaimeen.

5 Käyttöönotto

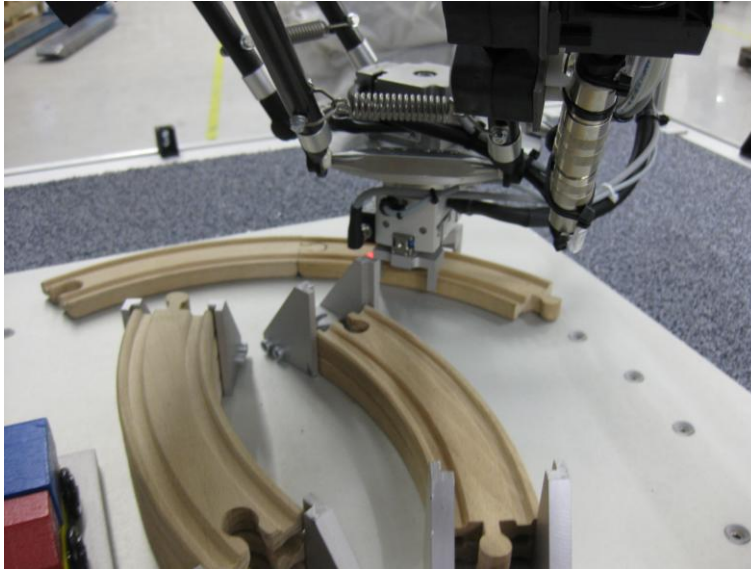
5.1 Toiminnan kuvaus

Projektin tavoitteena oli saada Tripod EXPT-robotti (kuva 36) rakentamaan puisista raiteista ympyränmuotoinen rata ja ajamaan siinä puista autoa. Ennen auton ajamista radalla robotti pyörii 3 kierrosta radan päällä näyttääkseen, kuinka nopea se on. Kun robotti on ajanut auton ja saanut radan purettua takaisin niille tarkoitettuihin makasiineihin, palaa tämä takaisin kotipisteeseen.

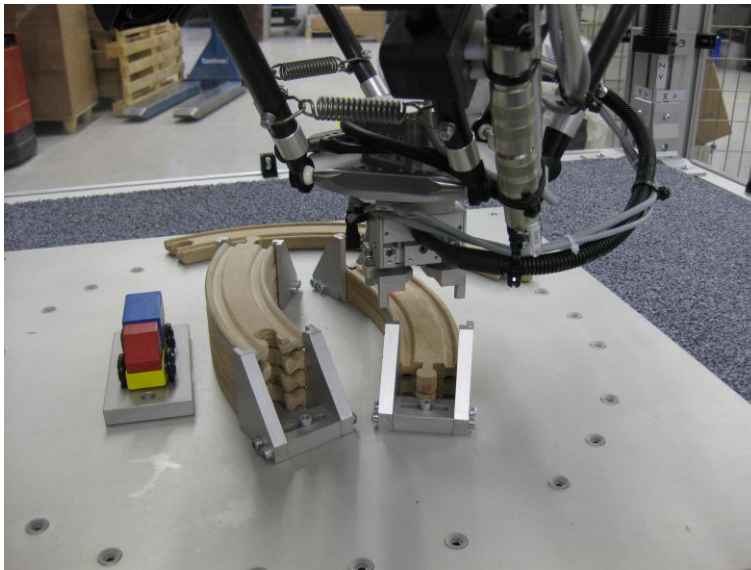


Kuva 36. Robotti kotipisteessä

Robotti hakee järjestyksessä maksiini 1:stä palikat 1, 2, 3 ja 4 ja asettaa nämä omille paikoilleen riippuen, mikä palikka on kyseessä (kuva 37 ja kuva 38).

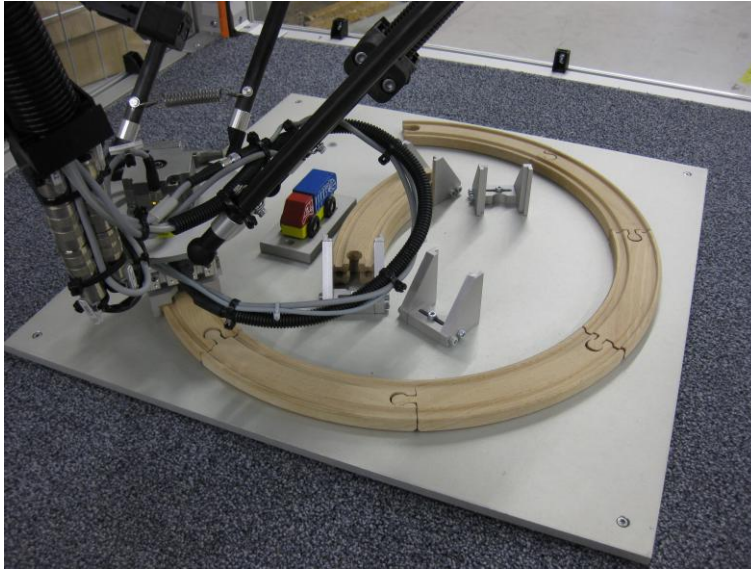


Kuva 37. Junaradan rakentaminen



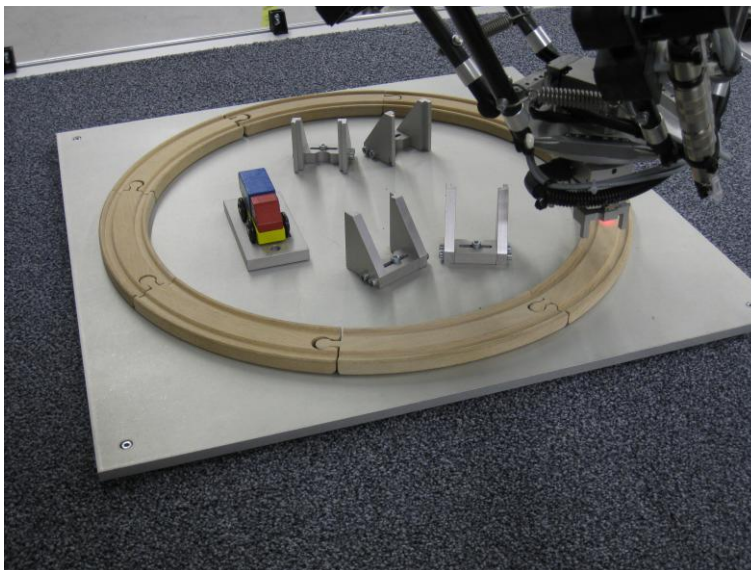
Kuva 38. Raiteen haku maksasiinista 1

Kasattuaan raiteen makasiini 1:stä ryhtyy robotti hakemaan raiteet 5, 6, 7 ja 8 makasiini kakkosesta ja viemään nämä niiden omille paikoille (Kuva 39)



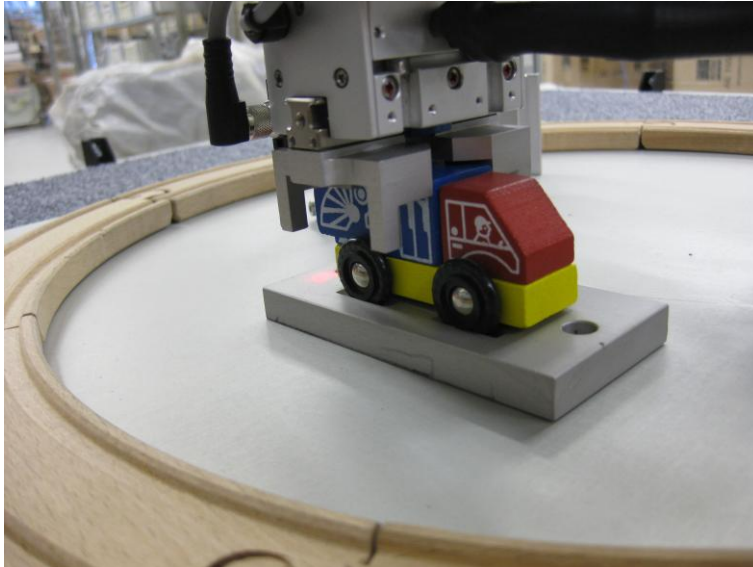
Kuva 39. Raiteiden kasaaminen makasiinista 2

Robotin saatua kasattua radan ajaa se tämän jälkeen kolme kierrosta ympyrää radan päällä (kuva 40). Tällä sovelluksella oli tarkoitus osoittaa, kuinka nopea robotti on, koska kyseissä radan kasausprojektissa ei käytetä riskisistä nopeampia nopeuksia.



Kuva 40. Ympyrää ajo

Ajettuaan ympyrää kolme kertaa hakee robotti puisen auton sen makasiinista (kuva 41). Tämän jälkeen robotti vie auton raiteen päälle ja vapauttaa sen raiteille (kuva 42).



Kuva 41. Auton haku sen makasiinista



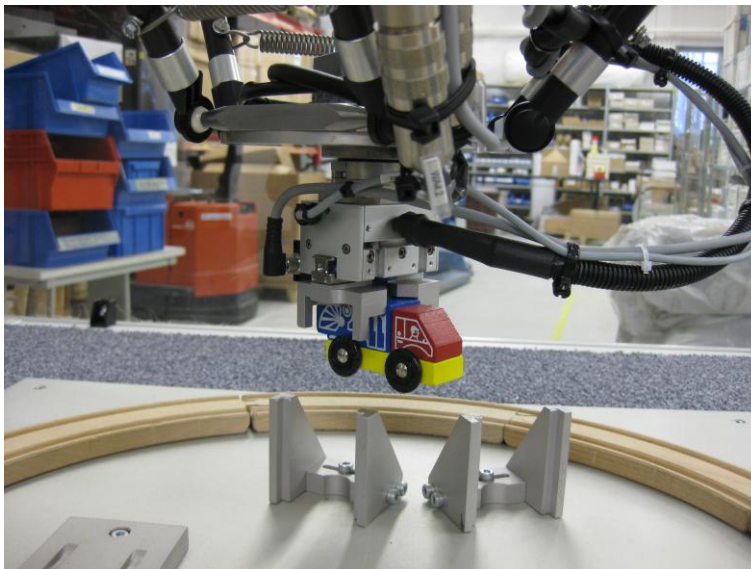
Kuva 42. Auton vapautus raiteille

Vapautettuaan auton robotti ryhtyy työntämään autoa rataa pitkin sen tarttujassa olevalla sormella (kuva 43).



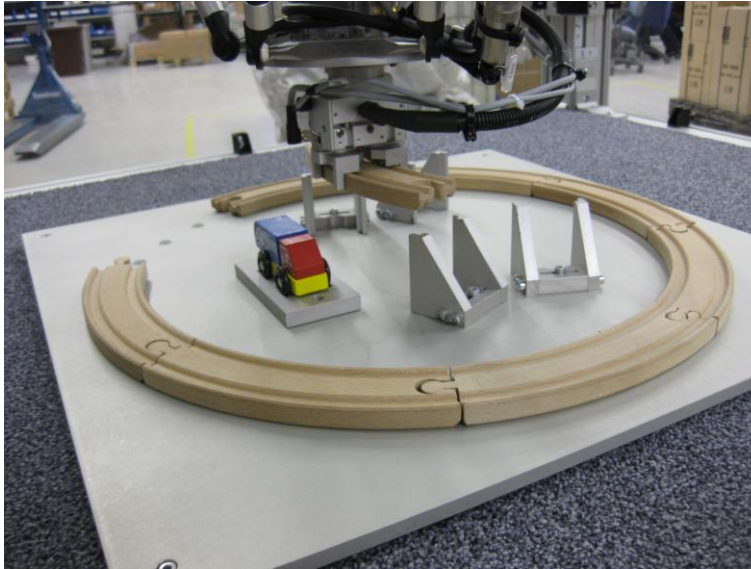
Kuva 43. Auton liikuttaminen raiteilla

Robotti vie auton takaisin sen makasiiniin, ajettuaan ensin tällä radan ympäri (kuva 44).



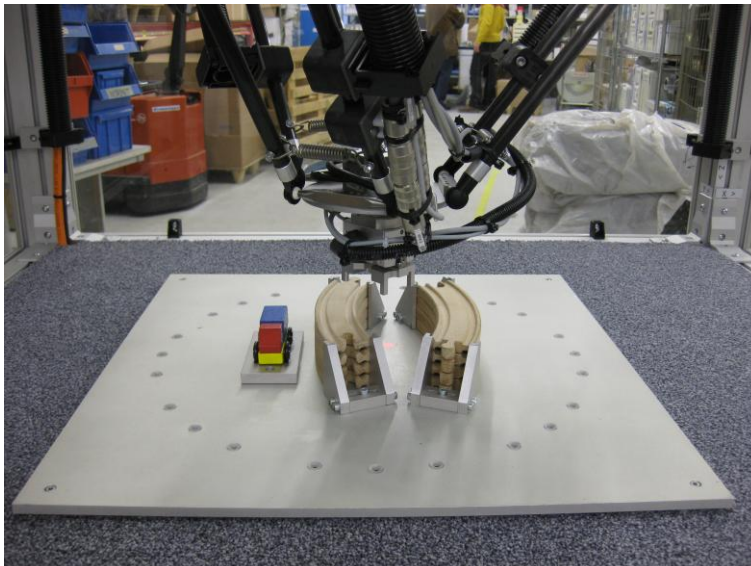
Kuva 44. Auton poimiminen radalta ja takaisinvienti sen makasiiniin

Lopuksi robotti purkaa radan makasiineihin 1 ja 2 päinvastaisessa järjestyksessä, mitä kasaussvaiheessa tapahtui (kuva 45).



Kuva 45. Junaradan purku takaisin makasiineihin

Kaiken suoritettuaan robotti palaa takaisin kotipisteeseensä odottamaan uutta käskyä toimia.



Kuva 46. Paluu kotipisteeseen

5.2 Ohjeiden laatiminen

Projektiin sisältyi myös käyttöohjeiden teko (Liite 4). Robottisovellukselle tehtiin yksinkertaiset ja selkeät ohjeet tämän käyttämisestä ja toiminnasta. Ohjeiden peruseriaate oli, että vähäisenkin perehdytyksen saanut henkilökin saisi ohjeiden avulla robotin toimimaan.

Ohjeet sisälsivät kuvalliset ohjeet vaihe vaiheelta sovelluksen käynnistämiseen, virhetilanteissa syntyvien virheviestien merkityksen sekä lyhyen ohjeistuksen niistä toipumiseen.

5.3 Testaus

Robottisovelluksesta haluttiin toimintavarma, minkä takia se testattiin tekemällä sille erilaisia virhetilanteita ja tarkastamalla, että robotti pysähtyy sellaisen sattuessa. Tyypillisiä virhetilanteita oli, jos ilmanpaine on liian alhainen, tarttuja ei vapauta raidetta, rata ei pysy paikallaan tai raide puuttuu.

Kun virhetilanteet oli testattu, suoritettiin robotille sadan kerran loop-käyttö eli robotti suorittaa radan kasauksen ja purkamisen sata kertaa.

Lopuksi Feston Tripod EXPT-robotin kanssa tekemisissä olleiden henkilöiden kanssa pidettiin palaveri, jossa käytiin läpi, kuinka hyvin projektiin sisältyvät tehtävät täyttyivät ja kuinka hyvin päästiin haluttuun tavoitteeseen. Lopuksi työ hyväksyttiin kyseisessä palaverissa.

5.4 Käyttöönotto asiakkaiden tiloissa

Lisätyönä projektiin kuului robotin käyttöönotto asiakkaiden tiloissa. Projektin käyttöönotto tapahtui Virossa Tallinnan teknillisessä yliopistossa. Käyttöönottoon kuului robottijärjestelmän toimintakuntoon laittaminen. Robottiin kytkettiin sähköt ja paineilma. Lisäksi robotille ajettiin kalibrointiohjelma, jolla pystyttiin testaamaan, että robotti ja työlevy eivät ole liikkuneet toisiinsa nähden kuljetuksen aikana.

Asiakkaalle annettiin kyseiseen robottisovellukseen tehty ohje, joka käytiin läpi yksityiskohtaisesti. Robotin käyttämiseen liittyviä seikkoja käytiin läpi käytännön harjoittelulla, jotta asiakas pystyisi ottamaan robotin uudelleen käyttöön sähköjen hävitessä.

6 Yhteenveto

Tripod EXPT Junarata -projekti sai alkunsa, kun robotin käytettävyys oli huono, koska kyseisessä laitteessa ei ollut esittelytilaisuuksiin sopivaa sovellusta. Robotti oli käyttämättömänä pitkään, ja tällä kyseisellä projektilla oli tavoitteena saada robotin hankintakustannuksille vastinetta sekä erilaisten esittelytilaisuuksien kautta näkyvyyttä Feston robottitarjonnasta.

Projekti aloitettiin tutustumalla robotin perustoimintoihin ja ohjelmointiin. Työ vaati runsaasti aikaa ja mielenkiintoa tutustua robotin manuaaleihin. Loppuen lopuksi pitkäjänteisyys palkittiin ja ohjelmointi alkoi sujua.

Työ muodostui kolmesta vaiheesta:

- suunnittelusta
- toteutuksesta
- käyttöönotosta.

Hyvä suunnittelu on projekteissa tärkeä osa. Hyvällä ja huolellisella suunnittelulla voidaan välttyä pahoilta virheiltä ja kustannuksilta. Suunnittelun ensimmäisessä vaiheessa pohdittiin ja analysoitiin työalustan eri kiinnitystapoja, minkä jälkeen pystyttiin suunnittelemaan ohjelman rakennetta. Toteutusvaiheessa suunnitteluihin tehtiin pieniä muutoksia, suurimmalta osin muutokset painoutuivat ohjelmointiin. Käyttöönotossa kaikki sujui hyvin, vaikka kyseinen projekti sisälsi paljon riskitekijöitä.

Projekti toteutui asetettujen tavoitteiden mukaisesti. Robottisovellus saatiin toimintavarmaksi sekä helppokäyttöiseksi. Robottia pystytään nykyisin viemään erilaisiin esittely- ja messutilaisuuksiin. Robotin käyttöä parannettiin ohjausnappulalla ja virhevalolla robottisolun päällä. Turvallisuutta parannettiin ylimääräisellä hätä-seis-katkaisimelle, sekä estämällä antureilla sovelluksen suoritusvaiheessa mahdollisesti syntyviä ristiriitoja.

Kyseisen projektin olisi voinut toteuttaa myös erilaisilla sovelluksilla, mutta kustannussyistä oli parempi toteuttaa kyseinen junaratasovellus. Puiset raiteet tuovat paljon riskitekijöitä toimivuuteen, koska useimpien toistojen ja ilmankosteusmuutosten myötä voi raiteiden muoto muuttua. Projektissa komponenttien valinnassa olisi voinut kiinnittää huomiota alipainetta tuottavien ejektoreiden ilman tarpeeseen. Projektiin valittiin kyseiset ejektorit niiden saatavuuden ja projektin etenemisen kannalta.

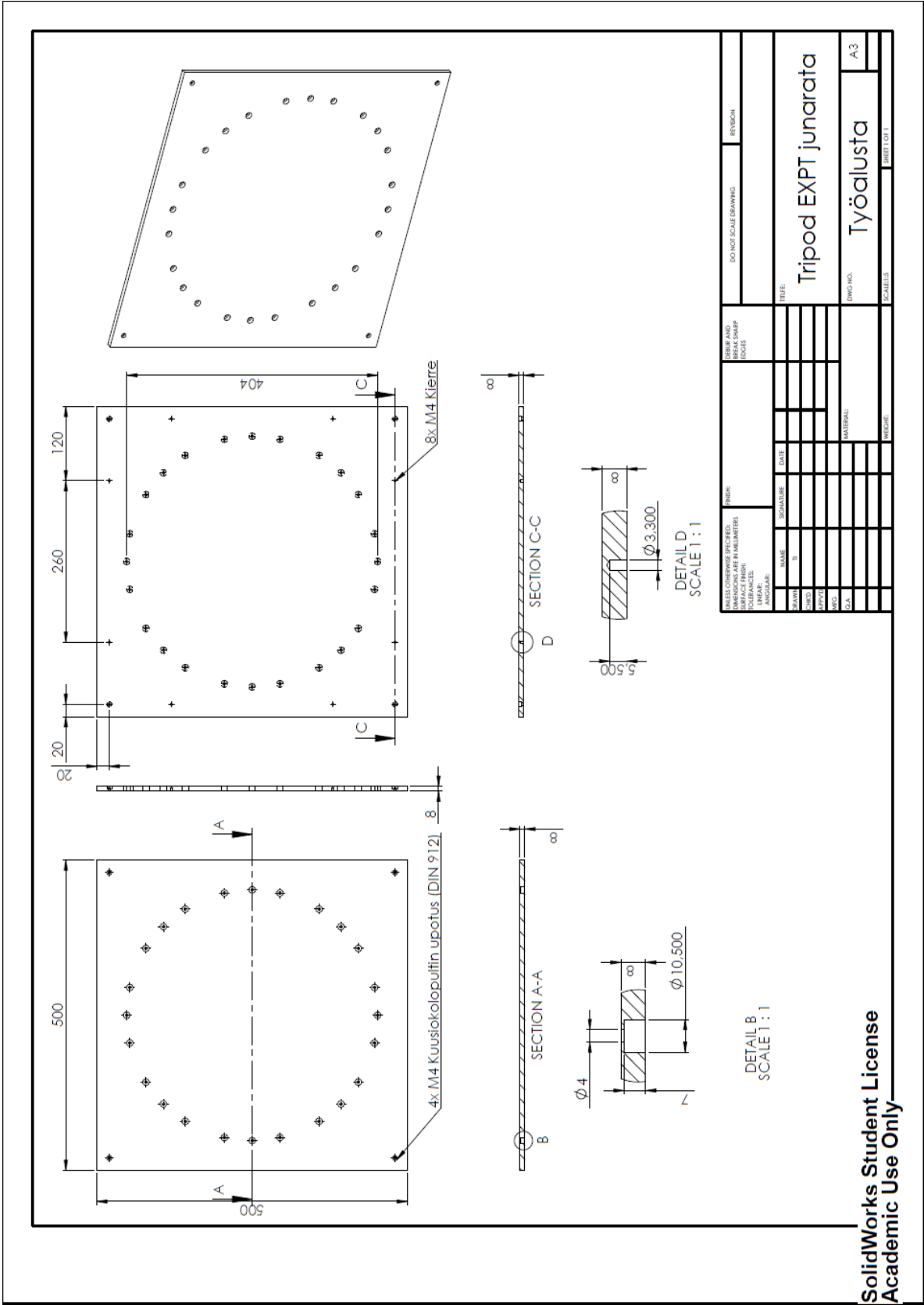
Työ opetti projektityöskentelytaitoja ja aikataulutusta, joka on mielestäni yksi tärkeimpiä asioita, kun halutaan saada työ tehtyä ilman katkoksia. Myös robotiikka tuli tutuksi, erityisesti rinnakkaisrakenteiset robotit ja se, mitä seikkoja näissä pitää ottaa huomioon. Työ kehitti mielestäni omia ohjelmointitaitojani, joita kyseisessä työssä vaadittiin paljon.

Lähteet

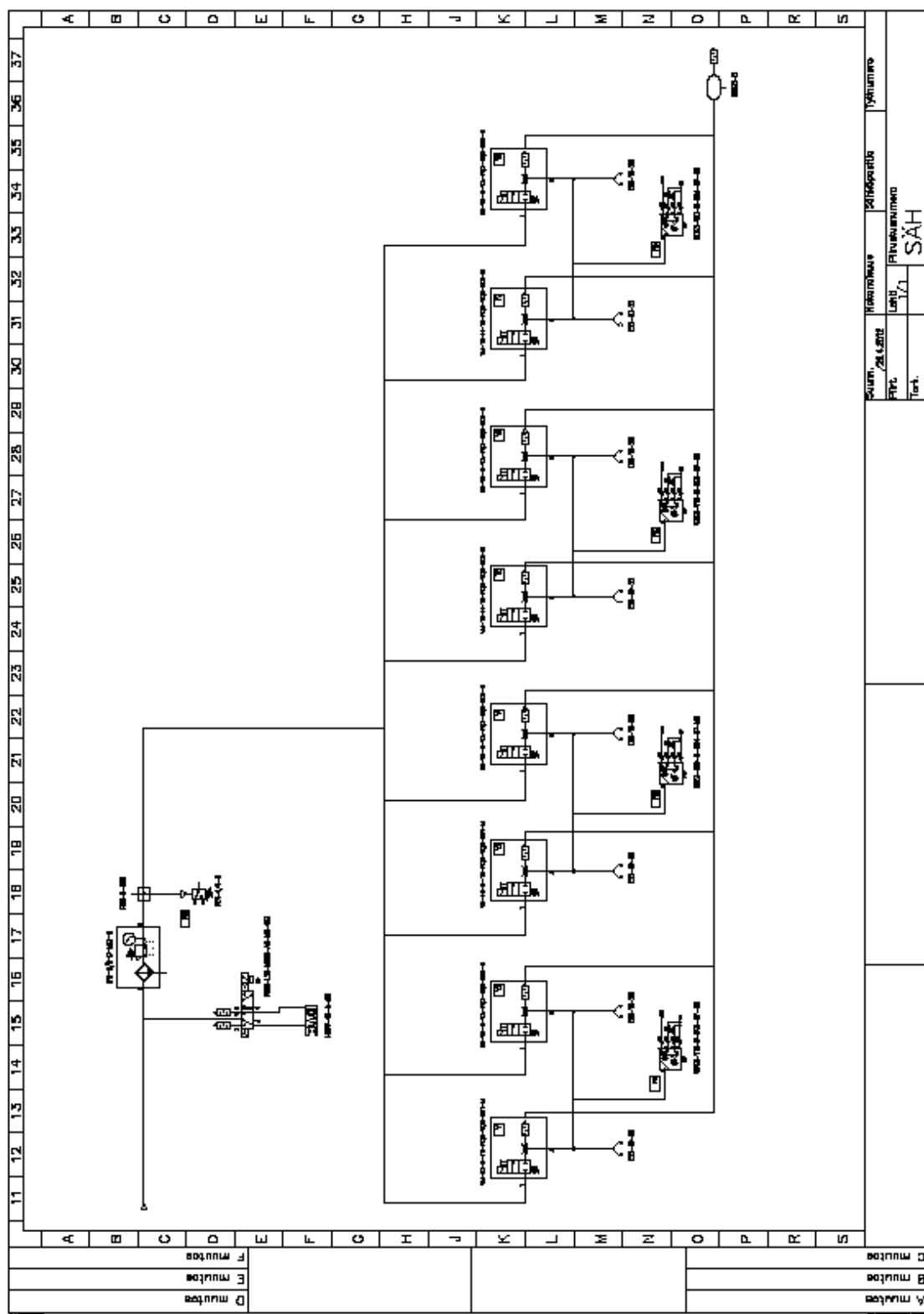
- 1 Festo kotisivu. Verkkodokumentti. Festo AG & Co KG. www.festo.fi. Luettu 26.12.2011
- 2 Robotiikka. 1999. Suomen Robotiikkayhdistys Ry. Tummavuoden Kirjapaino Oy. Vantaa
- 3 Jouni Jokelainen. Robotiikka -luentomoniste. PDF -dokumentti. Metropolia AMK Luettu 3.1.2012
- 4 SFS-ISO 10218-1. Teollisuusrobotit-Turvallisuusvaatimukset -robotti. Suomen Standardoimisliitto Ry.
- 5 Robotiikka.2008. Verkkodokumentti. Lahden Ammattikorkeakoulu. http://miniweb.lpt.fi/automaatio/opetus/luennot/pdf_tiedostot/Robotiikka_yleinen.pdf. Luettu 9.1.2012
- 6 Handling Guide. 2010. Verkkodokumentti. Festo AG & Co KG. http://www.festo.com/net/SupportPortal/Downloads/17200/Handling_Guide_en.pdf. Luettu 12.1.2012
- 7 Under control and in motion. Verkkodokumentti. Festo AG & Co KG. http://www.festo.com/net/SupportPortal/Downloads/17283/MotionControl_en.pdf. Luettu 1.2.2012
- 8 Teach pendant CDSA. Verkkodokumentti. Festo AG & Co KG. http://www.festo.com/net/SupportPortal/Downloads/13271/PSI_129_2_en.pdf. Luettu 2.2.2012
- 9 PSI_909_3_EXPT_en. Verkkodokumentti. Festo AG & Co KG. http://www.festo.com/net/SupportPortal/Downloads/17428/PSI_909_3_EXPT_en.pdf. Luettu 2.3.2012
- 10 Advance Handling Systems. Verkkodokumentti. Festo AG & Co KG. http://www.festo.com/net/SupportPortal/Downloads/131453/AdvancedHandlingSystems_en.pdf. Luettu 2.3.2012

- 11 GDCP-CMXR-HW-EN. 2008. Verkkodokumentti. Festo AG & Co KG.
<http://www.festo.com/net/SupportPortal/Downloads/52803/560322g1.pdf>.
Luettu 24.1.2012
- 12 Festo AG & Co KG. http://www.festo.com/cms/nl-be_be/17009.htm. Luettu 27.12.2011
- 13 GDCP-CMXR-SW-EN. 2005. Verkkodokumentti. Festo AG & Co KG.
<http://www.festo.com/net/SupportPortal/Downloads/52788/560316g1.pdf>.
Luettu 27.12.2011.
- 14 Handling_Guide_en. Verkkodokumentti. Festo AG & Co KG.
http://www.festo.com/net/SupportPortal/Downloads/17200/Handling_Guide_en.pdf.
Luettu 27.12.2011.
- 15 GDCP-CMXR-SY-EN. 2009. Verkkodokumentti. Festo AG & Co KG.
<http://www.festo.com/net/SupportPortal/Downloads/52778/560310g1.pdf>.
Luettu 13.3.2012.

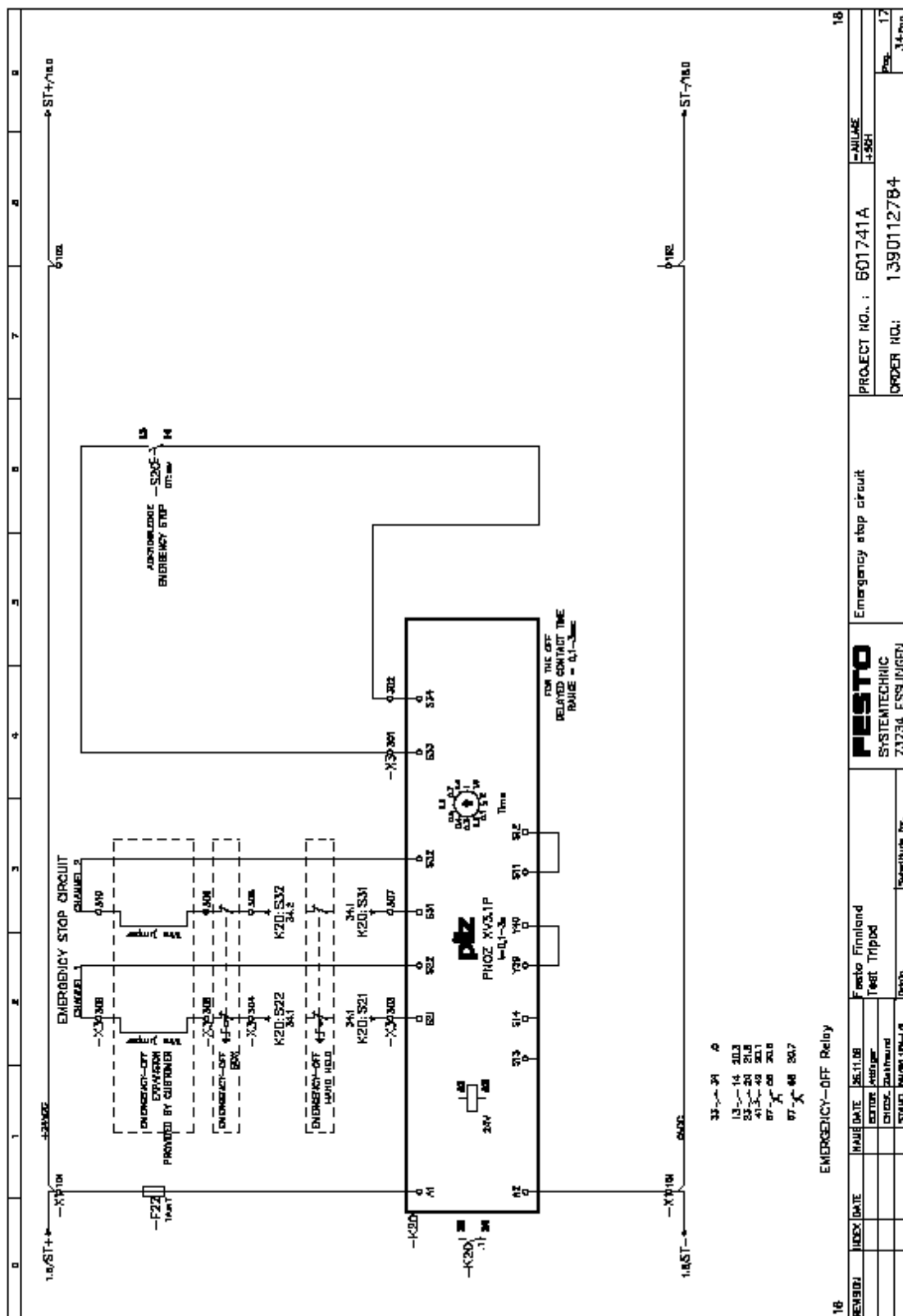
Työalustan layout

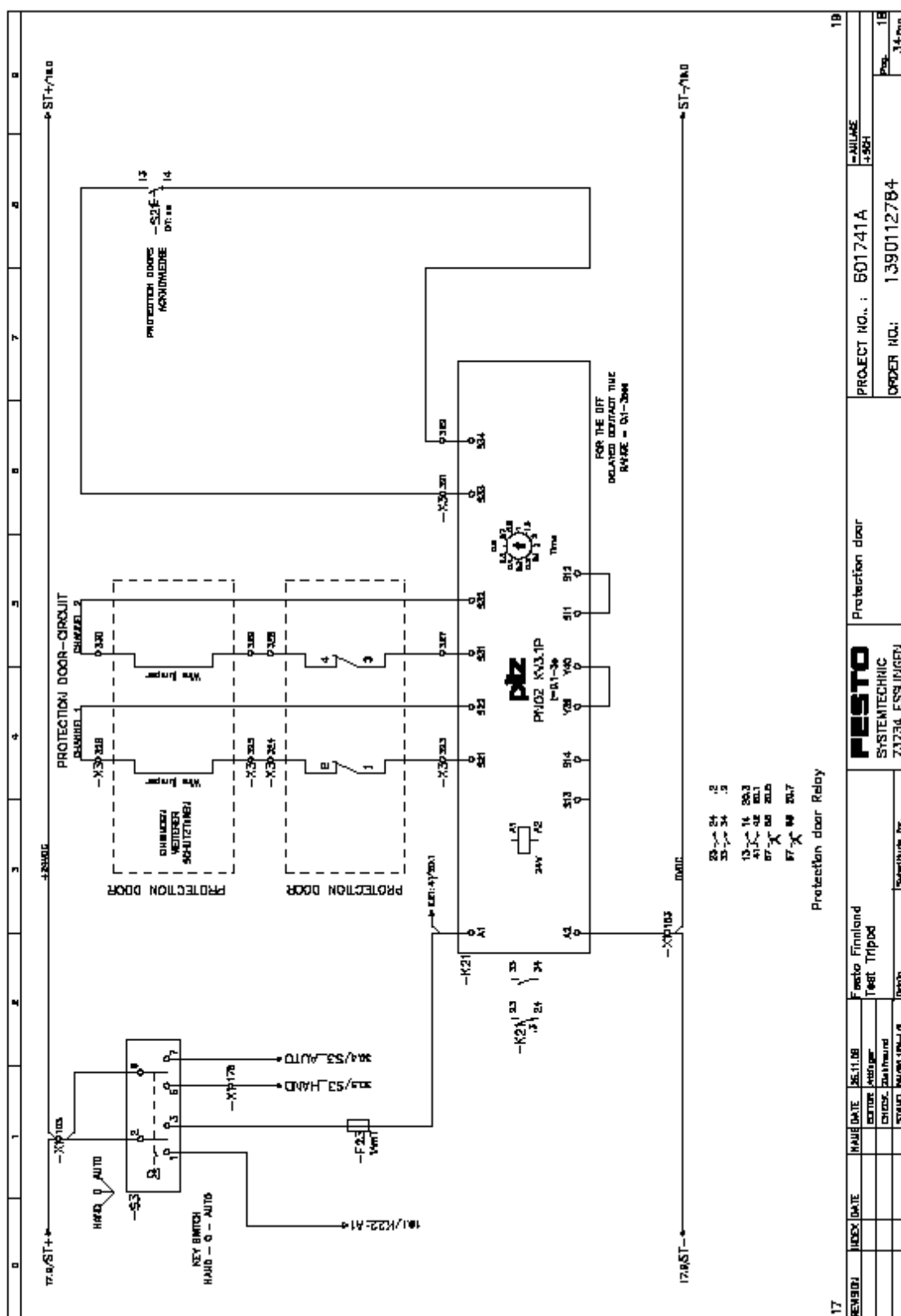


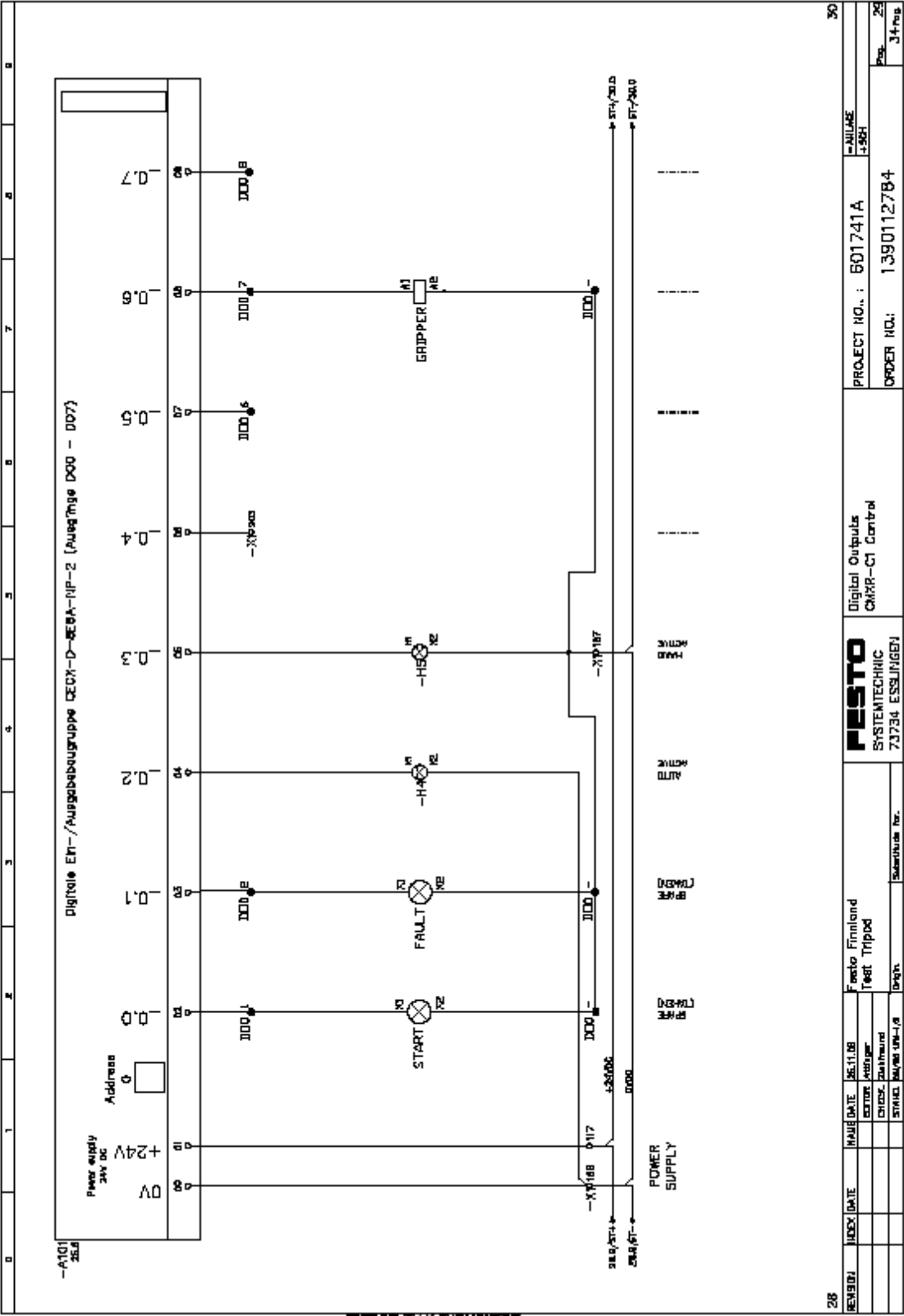
Pneumatiikkakuva

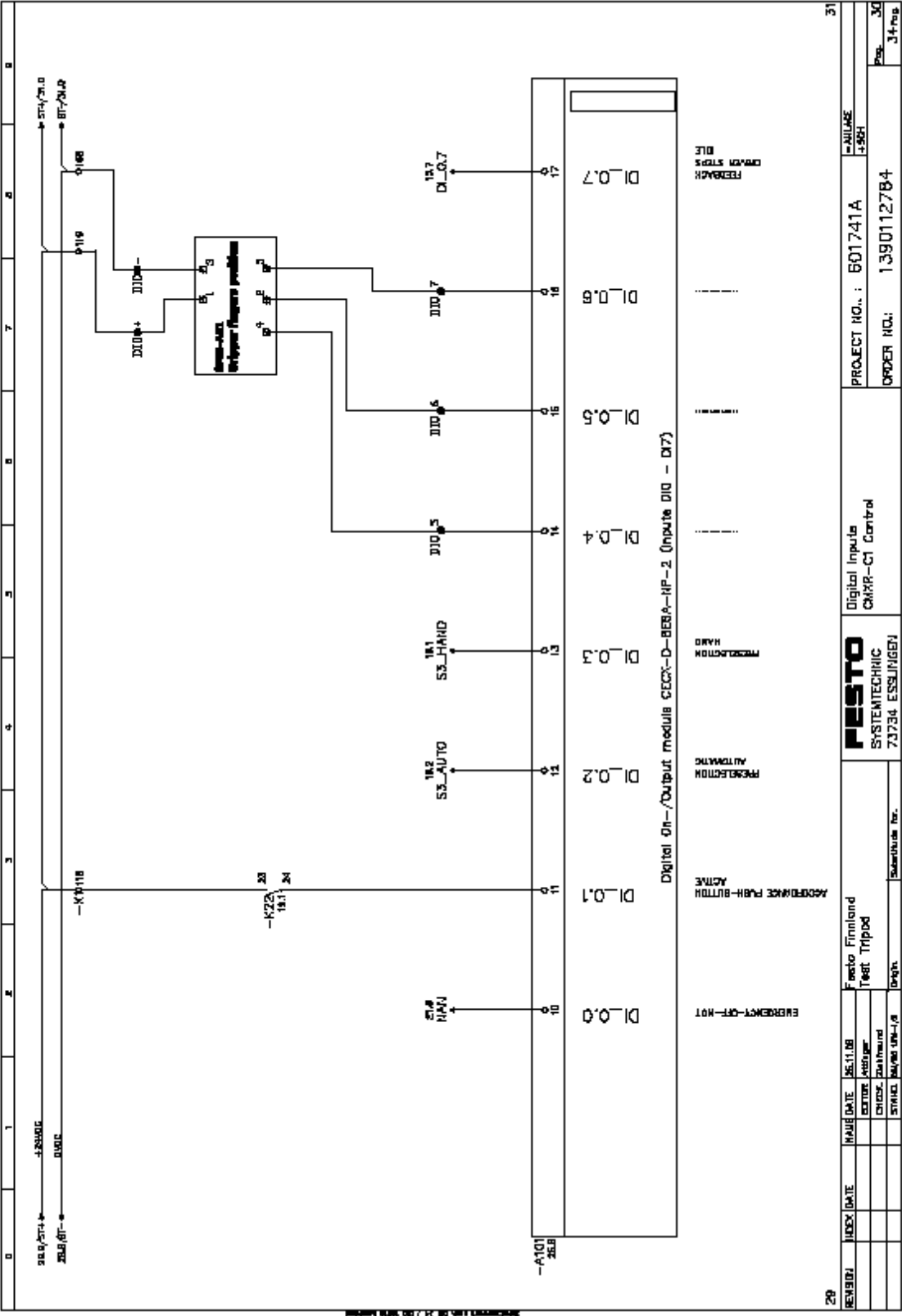


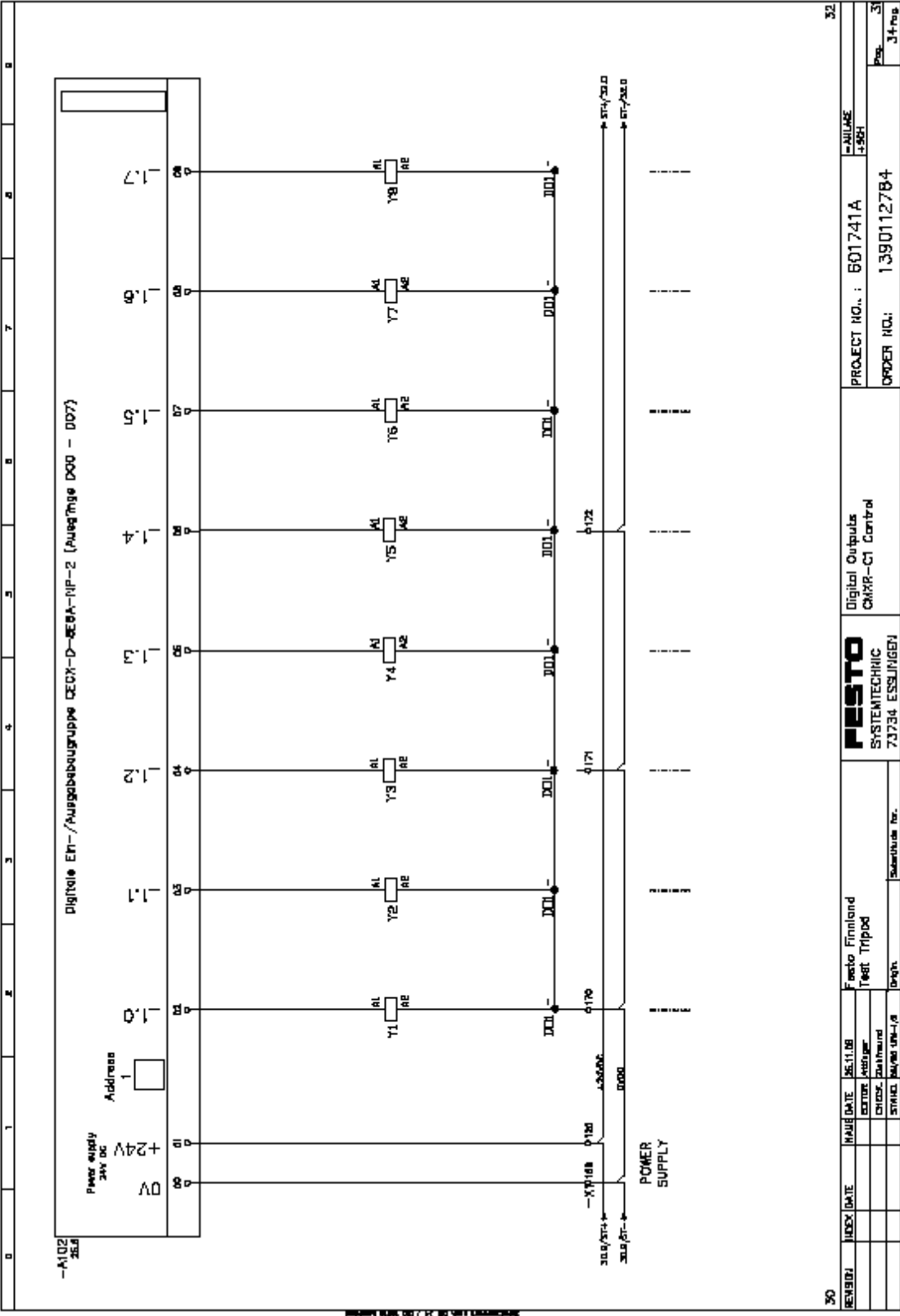
Liite 3
1 (7)

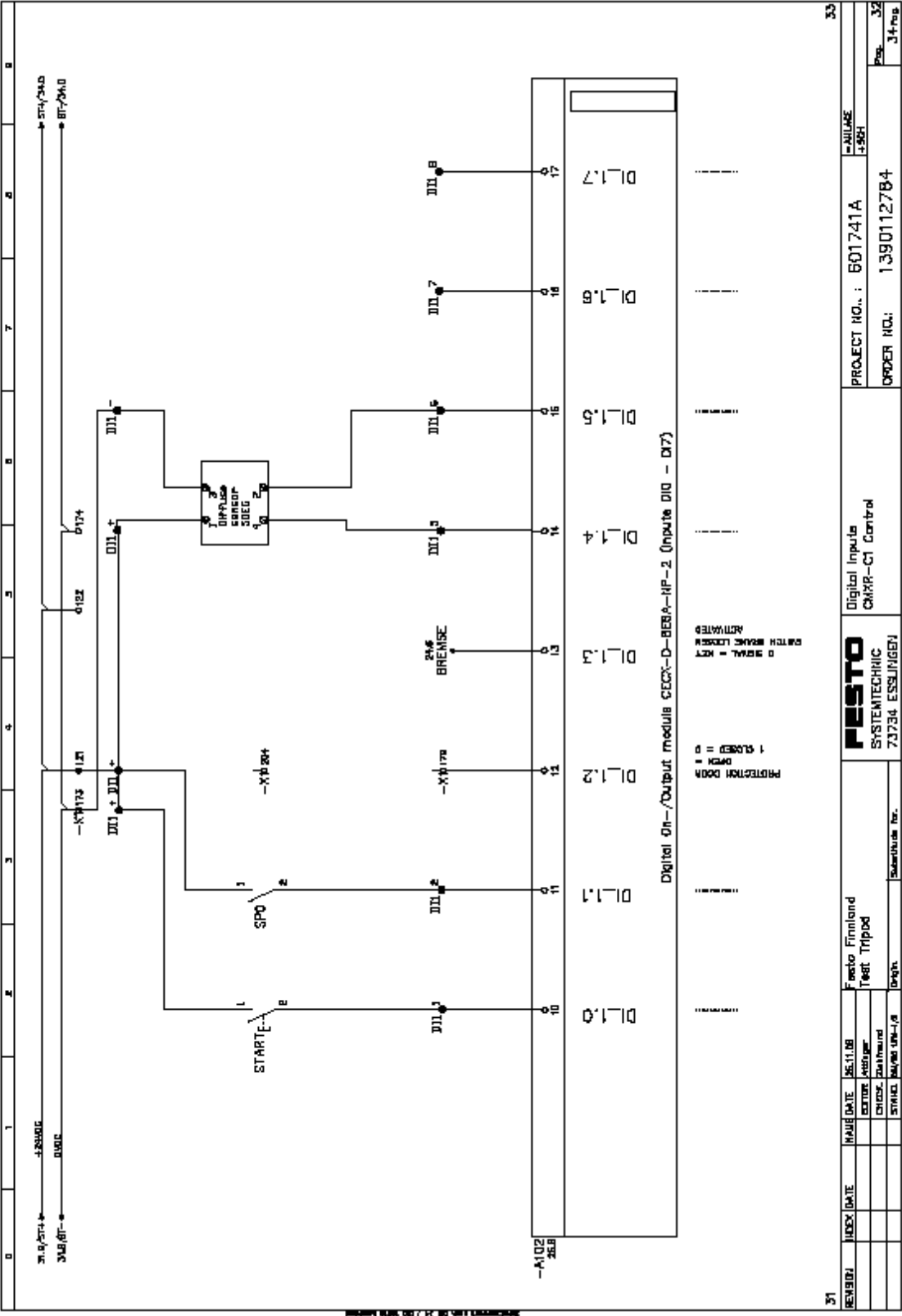


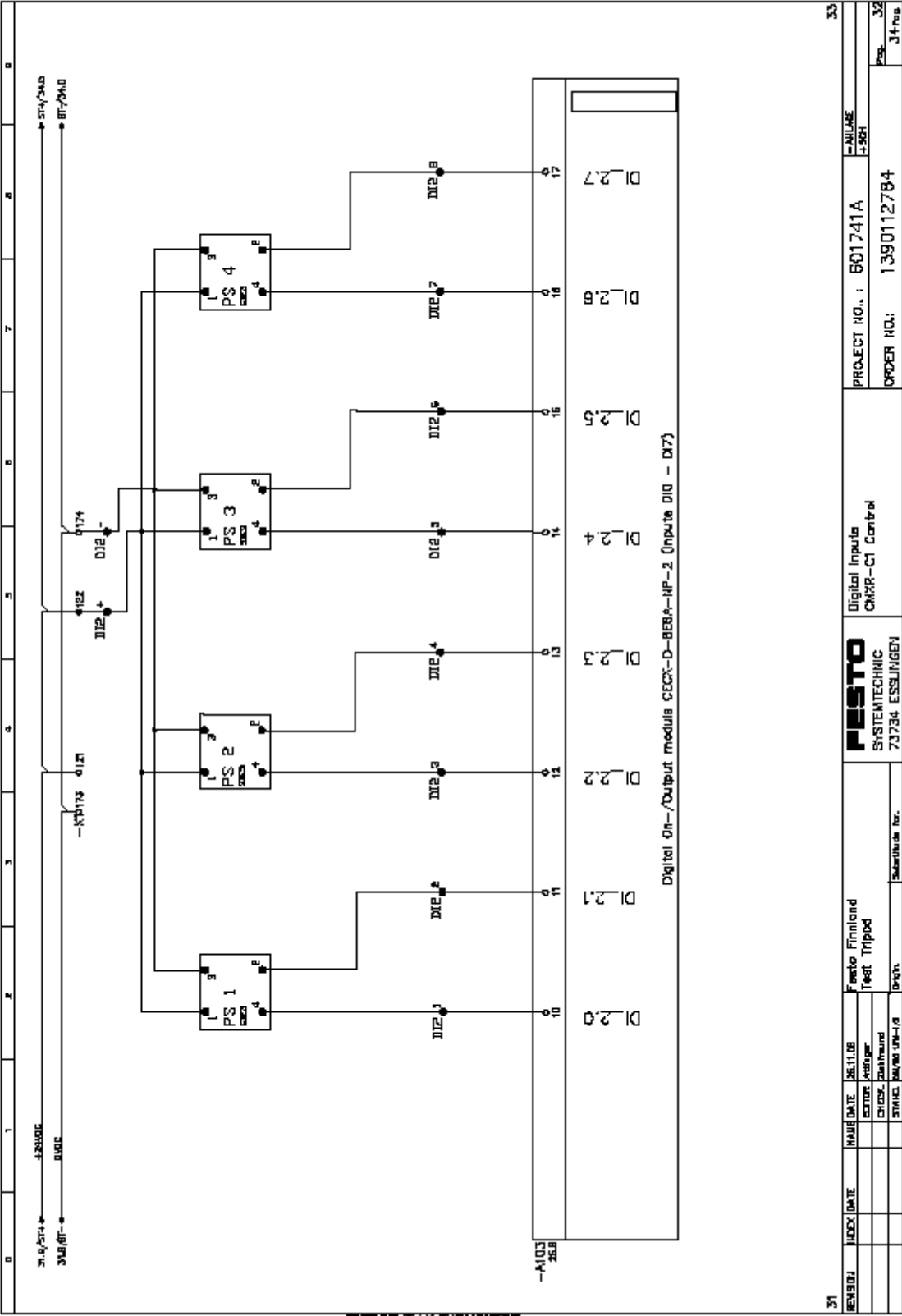












Käyttöohje

FESTO TRIPOD EXPT JUNARATASOVELLUKSEN-KÄYTTÖOHJE

Tripod-käyttöohje

26.1.2011

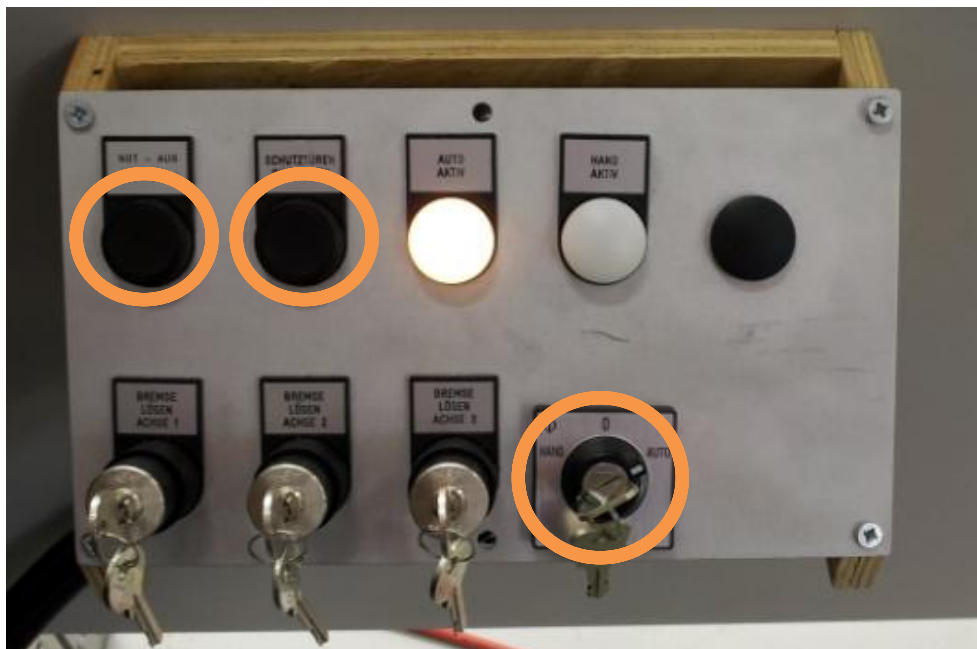
Sisällys

1 Käyttöönottoohje	1
2 Robotin kalibrointi.....	4
3 Häiriöviestit.....	6

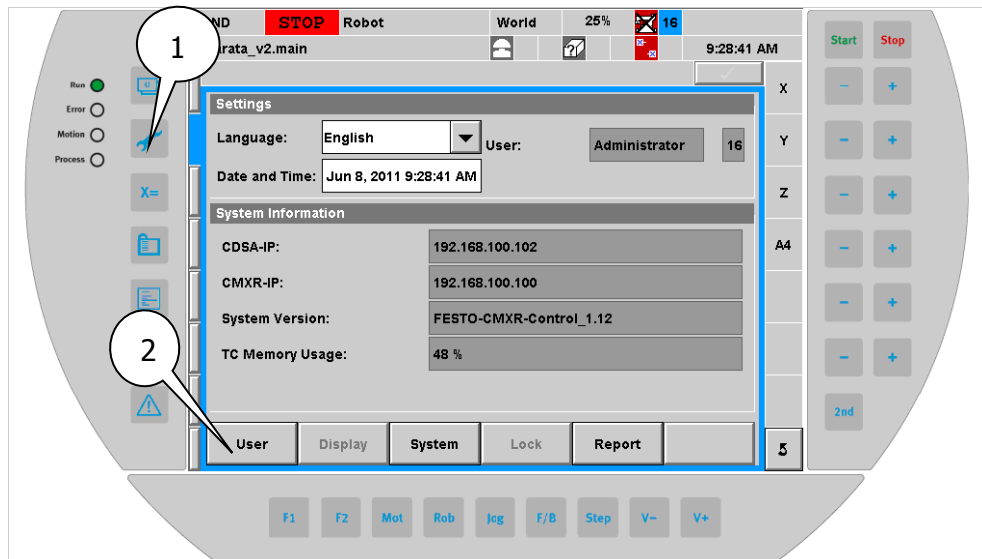
Liite 1 Junaradan raiteiden ja junan paikat makasiineissa

1 Käyttöönotto-ohje

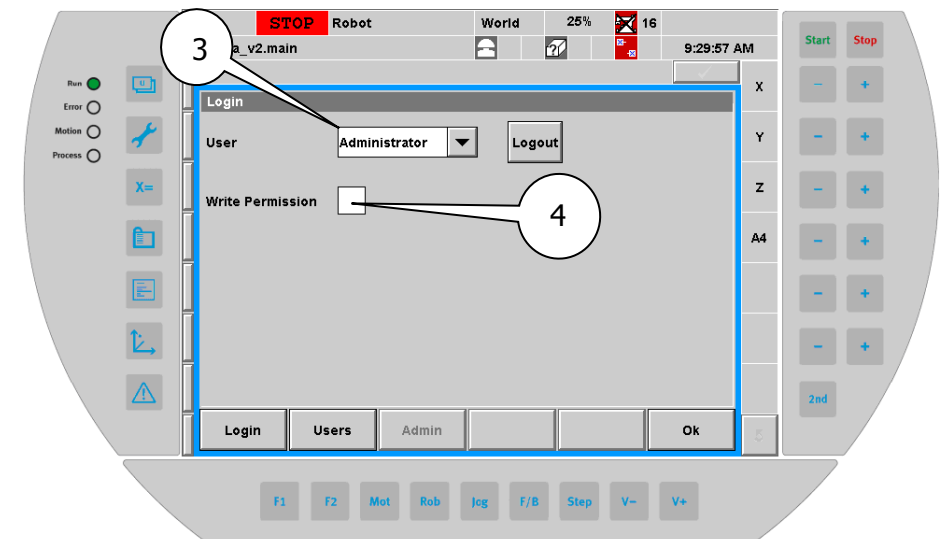
1. Kytke voimapistoke seinään sekä kytke laitteeseen paine.
2. Lado junaradan palikat ja juna oikeaan järjestykseen makasiineihin (Liite 1)
3. Käännä pääkytkin kaapin sisällä I -asentoon ja odota, että järjestelmä käynnistyy.
4. Kun järjestelmä on käynnistynyt, käännä kaapin ovesta oleva Auto / Hand-kytkin asentoon Auto. Paina kahta mustaa nappia ja toista tämä, jos avaat lasioven tai painoit hätä-seis-katkaisinta.



5. Paina työkalunappia käsipaneelista (1), jonka jälkeen User softkey -nappia (2) alhaalta.



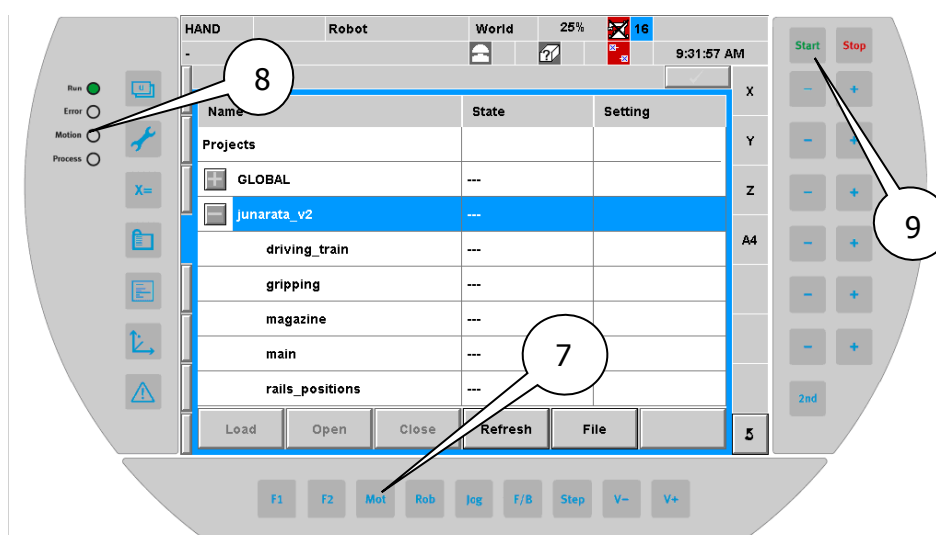
6. Valitse User (3) kohdasta **Administrator** ja syötä salasanaksi: **admin** ja valitse Write Permission (4). Jatka painamalla Ok.



7. Paina vasemmalta kansionappia (5), jonka jälkeen pitäisi näkyä valittavat ohjelmat. Klikkaa listasta **junarata_v2** ja valitse aliohjelma **main**. Paina Load softkey -nappia (6).



8. Ruudulla pitäisi näkyä Main-ohjelma. Paina alhaalta Mot-nappia (7). Vasemmalla pitäisi nyt palaa Motion-ledi (8) ja ylhäällä lukea: Robot activated.
9. Start-nappia (9). Nyt robotin pitäisi ruveta suorittamaan ohjelmaa. Ohjelman voi pysäyttää painamalla Stop-nappia.



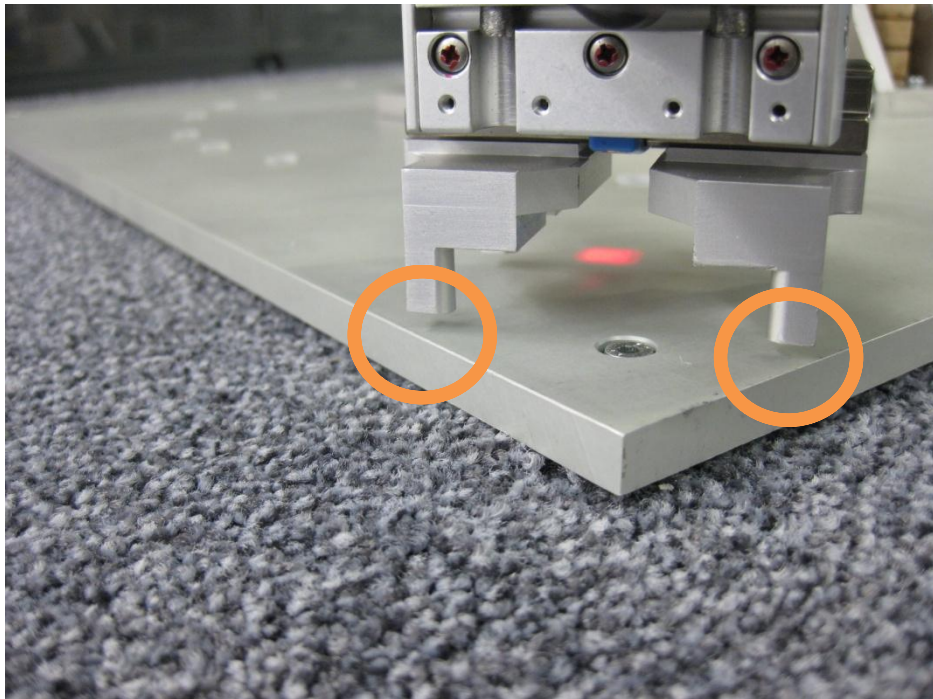
HUOM!

Mikäli robotti ei käynnisty ja punainen valo palaa, katso ohjeiden kohta Häiriöviestit.

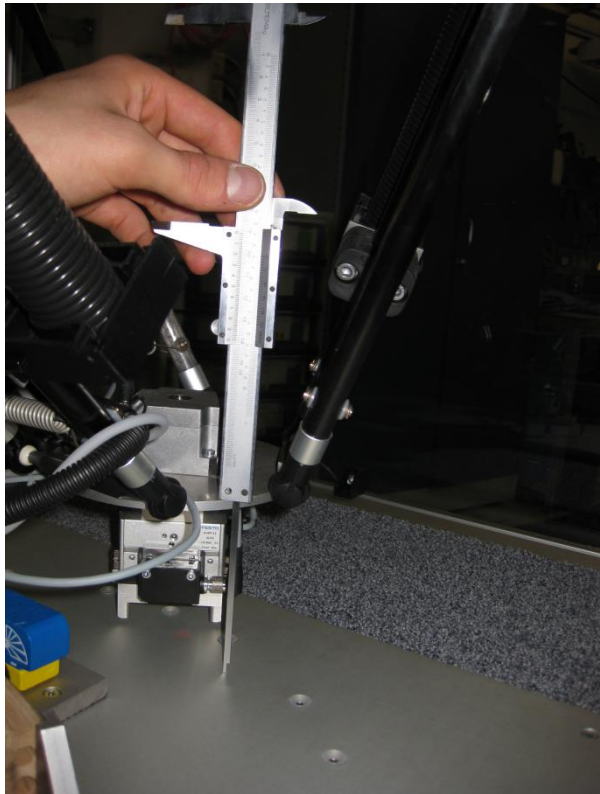
2 Robotin kalibrointi

Mikäli robotti otetaan kuljetuksen jälkeen käyttöön, tulee tämä kalibroida kyseisen ohjeen mukaan.

1. Käyttöönota robotti Käyttöönotto-ohjeiden mukaisesti paitsi kohdassa 6 valitset aliohjelma calibration.
2. Robotti suorittaa Calibration-ohjelman loopin 3 kertaa. Tarkasta, että robotin tarttujan "kynnet" menevät levyn reunan mukaisesti, ja säädä levyä tarvittaessa.



3. Lopuksi robotti pysähtyy ja pyytää tarkistamaan robotin z-akselin korkeuden. Tarkista korkeus työntömitalla kuvan mukaisesti. Arvon tulisi olla 91 mm



4. Mikäli kaikki kalibrointi-arvot ovat kohdallaan, paina vihreää nappia robotin päällä 5 s, jolloin robotti käynnistää Main-ohjelman ja robotti on valmis käyttöön.

HUOM!

Mikäli robotti ei käynnisty ja punainen valo palaa, katso ohjeiden kohta Häiriöviestit.

3 Häiriöviestit

Jos robotti ei käynnisty tai pysähtyy kesken ohjelman, niin tarkasta seuraavat asiat:

1. Onko robotin ovi kiinni ja ovilukko kuitattu?
2. Ovatko hätä-seis -napit ylhäällä ja kuitattu?
3. Palaako punainen merkkivalo? Jos valo palaa, tarkasta robotin käsipaneelista häiriöviesti ja katso seuraavasta listasta, miten häiriöstä toivutaan.



Häiriöviesti: **Check pressure**

Toipuminen: Mikäli kyseinen häiriö tulee robotin käyttöönotossa, voit toipua tästä syöttämällä paineen laitteeseen, jolloin häiriövalon pitäisi sammua ja vihreän napin syttyä palamaan.

Jos virhe tulee ajon aikana, lada junaradat takaisin makasiiniin ja lataa ohjelma uudestaan.

Häiriöviesti: **Rail is not in position**

Toipuminen: Tarkista, onko robotti vienyt junaradan oikeaan paikkaan ympyrässä, ja lada junaradat makasiiniin ja lataa ohjelma uudestaan.

Häiriöviesti: **Rail is missed**

Toipuminen: Tarkista, puuttuuko junaradan palikoita tai ovatko ne oikeassa paikassa, ja lada junaradat makasiiniin ja lataa ohjelma uudestaan.

Häiriöviesti: **Rail is stack**

Toipuminen: Tarkista, onko laitteessa paineita ja onko tarttuja auki. Tämän jälkeen lada junaradat makasiiniin ja lataa ohjelma uudestaan.

Häiriöviesti: **No vacuum**

Toipuminen: Tarkista, puuttuuko junaradan palikoita, ovatko junaradan palikat oikeilla paikoillaan tai onko paine liian alhainen. Tämän jälkeen lada junaradat makasiiniin ja lataa ohjelma uudestaan.

Häiriöviesti: **Gripper is closed**

Toipuminen: Tarkista, onko laitteessa paineita ja onko tarttuja auki. Tämän jälkeen lada junaradat makasiiniin ja lataa ohjelma uudestaan.

